

PROPUESTA DEL CONTENIDO PARA UN PRIMER CURSO DE MATEMÁTICAS COMPUTACIONALES EN INGENIERÍA DE SISTEMAS

Ing. HUGO HUMBERTO MORALES PEÑA

MAESTRÍA EN ENSEÑANZA DE LAS MATEMÁTICAS
Línea de Matemáticas Computacionales

UNIVERSIDAD TECNOLÓGICA DE PEREIRA
FACULTAD DE CIENCIAS BÁSICAS
DEPARTAMENTO DE MATEMÁTICAS
Pereira, Risaralda
25 de Febrero de 2011

Índice general

1. Introducción	5
1.1. Problema objeto de estudio	5
1.2. Objetivos de la Investigación	5
1.2.1. Objetivo General	5
1.2.2. Objetivos Específicos	5
2. Estado del arte	7
2.1. Ingeniería de Sistemas a nivel mundial	7
2.1.1. La Ingeniería de Sistemas	8
2.1.2. Diferentes disciplinas de Ingeniería de Sistemas a nivel mundial	8
2.1.3. Representación gráfica de los tópicos de las disciplinas de Ingeniería de Sistemas	11
2.2. Libros de matemáticas computacionales	14
2.2.1. Estructuras de Matemáticas Discretas para la Computación - Kolman, Busby y Ross	15
2.2.2. Matemática Discreta - Norman L. Biggs	15
2.2.3. Matemáticas Discreta y Combinatoria - Ralph Grimaldi	15
2.2.4. Matemáticas Discretas - Ross y Wright	15
2.2.5. Matemáticas Discretas - Richard Johnsonbaugh	15
2.2.6. Matemáticas Discretas y sus Aplicaciones - Kenneth Rosen . . .	16
2.2.7. Matemáticas para la Computación - José A. Jiménez Murillo . .	16
2.3. Material de apoyo en universidades extranjeras	17
2.3.1. MIT - Massachusetts Institute of Technology - Estados Unidos .	17
2.3.2. Princeton University - Estados Unidos	18
2.3.3. Universidad de Málaga - España	18
2.3.4. Universidad Simón Bolívar - Venezuela	19
2.3.5. Instituto Tecnológico de Nuevo Laredo - Mexico	19
2.4. Material de apoyo en universidades colombianas	20
2.4.1. Universidad de los Andes - Bogotá	20
2.4.2. Universidad EAFIT - Medellín	21
2.4.3. Universidad ICESI - Cali	21
2.4.4. Universidad del Valle - Cali	22
2.4.5. Universidad Nacional - Bogotá	23
2.4.6. Pontificia Universidad Javeriana - Cali	23
2.4.7. Universidad de Antioquia - Medellín	24

2.4.8. Universidad del Norte - Barranquilla	25
2.5. Contenido de matemáticas computacionales	25
2.6. Exámenes de calidad en la educación superior	28
3. Asignaturas en matemáticas computacionales	31
3.1. Programa de matemáticas computacionales I	31
3.2. Programa de matemáticas computacionales II	39
3.3. Programa de matemáticas computacionales III	46
3.4. Programa de matemáticas computacionales IV	54
4. Estrategia construcción material de apoyo	63
4.1. El trabajo con sucesiones (o series)	63
4.2. El trabajo con la serie geométrica	64
4.3. El trabajo con sumatorias	65
4.4. Uso de sumatorias en solución de sumatorias	67
4.5. Uso de inducción matemática como método validador	69
5. Construcción banco de preguntas tipo ECAES	73
6. Resultados esperados	77
7. Conclusiones	79
8. Sugerencias y trabajos futuros	81
A. Manual de construcción de preguntas tipo ECAES	87
A.1. Aspectos técnicos	87
A.2. Qué y cómo se evalúa	87
A.3. Reglas para construir preguntas	90

Capítulo 1

Introducción

1.1. Problema objeto de estudio

El avance en el desarrollo de las ciencias y la tecnología involucra cada vez más teorías matemáticas; esto hace que sea prácticamente imposible incluir toda esta matemática en el contenido curricular de un programa de pregrado. Por lo tanto es necesario, teniendo en cuenta que en este momento se esta llevando a cabo una profunda reforma curricular en la carrera de Ingeniería de Sistemas y Computación de la Universidad Tecnológica de Pereira, hacer un estudio para determinar cuales son los contenidos básicos en el área de la matemática computacional que deben estar en el currículo, teniendo en cuenta principalmente el perfil del ingeniero de sistemas que se quiere para esta institución. Actualmente hay varios textos y muchos sitios en la red relacionados con matemáticas computacionales, pero, una vez determinados los contenidos básicos, es necesario enfocar al estudiante en estos temas mediante una guía adecuada que le permita obtener un proceso exitoso de aprendizaje.

1.2. Objetivos de la Investigación

1.2.1. Objetivo General

Determinar cuales son los temas fundamentales en Matemáticas Computacionales que deben formar parte de la malla curricular para la formación del ingeniero de sistemas de la Universidad Tecnológica de Pereira, según el perfil profesional determinado por el comité curricular. Y, diseñar los programas de las asignaturas y desarrollar un material de apoyo para la enseñanza del primer curso del área de Matemáticas Computacionales.

1.2.2. Objetivos Específicos

- Hacer la recolección de componentes, temas y subtemas que según los marcos de fundamentación conceptual del Ministerio de Educación nacional, debe formar parte de los cursos de matemáticas computacionales, en un programa de Ingeniería de Sistemas.

- Diseñar los programas de las materias que corresponden al área de Matemáticas Computacionales en el plan de estudios de Ingeniería de Sistemas.
- Hacer las guías de clase del primer curso del área de Matemáticas Computacionales.
- Crear un banco de preguntas de preguntas tradicionales y tipo ECAES para el primer curso del área de Matemáticas Computacionales.

Capítulo 2

Estado del arte

En este capítulo se presenta una visión general de la Ingeniería de Sistemas y de sus matemáticas computacionales desde un contexto mundial al contexto nacional, se pretende ir de lo general a lo particular desde el punto de vista del estado del arte en el cual esta enmarcada esta investigación. Por este motivo en la sección 2.1 se presentan las diferentes disciplinas de la Ingeniería de Sistemas a nivel mundial, luego, en la sección 2.2 se presentan algunos de los libros clásicos a nivel mundial de matemáticas discretas, donde se presenta un análisis del material de apoyo para la enseñanza que ellos brindan. En la sección 2.3 se presenta una revisión de universidades extranjeras con material de apoyo para la enseñanza de las matemáticas computacionales, de forma análoga en la sección 2.4 se hace la misma revisión de universidades pero en el contexto colombiano. Por último, en las secciones 2.5 y 2.6 se presenta el contenido que el Ministerio de Educación Nacional Colombiano sugiere para las matemáticas computacionales en los programas de Ingeniería de Sistemas y algunas directrices sobre la ley y el decreto que reglamentan los Exámenes de Calidad en la Educación Superior (ECAES) en Colombia, respectivamente.

2.1. Ingeniería de Sistemas a nivel mundial

Esta sección fue escrita tomando como base el documento “Computing Curricula 2005, The Overview Report covering undergraduate degree programs in: Computer Engineering, Computer Science, Information Systems, Information Technology and Software Engineering”, documento generado como el estándar a nivel mundial de los diferentes enfoques que tiene la Ingeniería de Sistemas en el mundo. Este currículo de computación fue generado de forma cooperativa por tres de las más prestigiosas asociaciones a nivel mundial en computación; ellas son ACM (Association for Computing Machinery), AIS (Association for Information Systems) y IEEE-CS (Institute for Electrical and Electronic Engineers - The Computer Society). Este documento es referenciado en la bibliografía como [ACMIEEE2005].

2.1.1. La Ingeniería de Sistemas

El termino “Ingeniería de Sistemas” es utilizado en Colombia para referirse al termino en inglés “Computing” (computación). La Ingeniería de Sistemas incluye el diseño y construcción de sistemas de hardware y software para una gran variedad de usos; estructuración, almacenamiento y procesamiento de diferente tipo de información; uso del computador para investigación científica; construcción de soluciones de software inteligentes; creación de aplicaciones multimedia; búsqueda y recopilación de información relevante, entre otros.

Tradicionalmente, un estudiante de Ingeniería de Sistemas es formado como profesional en una sola de las diferentes disciplinas de la Ingeniería de Sistemas, esto es debido a que la Ingeniería de Sistemas brinda una gran cantidad de posibilidades, y sería imposible que el estudiante sea competente en todas. Lo que se requiere es que de forma particular los programas de Ingeniería de Sistemas se enfoquen en la disciplina de Ingeniería de Sistemas en la cual quiere que sus egresados se desempeñen en su vida profesional.

Hay actualmente cinco disciplinas principales para los programas de pregrado en Ingeniería de Sistemas; cada una provee diferentes enfoques de la Ingeniería de Sistemas. En la siguiente subsección, se presentarán estas diferentes disciplinas de la Ingeniería de Sistemas y como ellas se comparan en términos de su enfoque y de la clase de problemas que resuelven.

2.1.2. Diferentes disciplinas de Ingeniería de Sistemas a nivel mundial

Ingeniería Computacional (Computer Engineering)

La Ingeniería Computacional se enfoca en el diseño y construcción de computadores y sistemas basados en el computador. Esto involucra el estudio de hardware, software, comunicaciones y la interacción entre ellos. Su currículo se enfoca en la teorías, principios y practicas de la Ingeniería Eléctrica tradicional y de las matemáticas y sus aplicaciones a los problemas de diseño de computadores y dispositivos basados en el computador.

Los estudiantes de Ingeniería Computacional estudian el diseño de sistemas de hardware digital que incluye sistemas de comunicaciones, computadores y dispositivos que contienen computadores. Ellos estudian desarrollos de software que se enfocan en dispositivos digitales y sus interfaces con usuarios y otros dispositivos. La Ingeniería Computacional le da prioridad al hardware sobre el software.

Actualmente, una gran área de desempeño de la Ingeniería Computacional son los sistemas embebidos, el desarrollo de dispositivos que tienen software y hardware embebidos en ellos. Por ejemplo, dispositivos tales como teléfonos celulares, reproductores

de audio digital, cámaras de video digital, sistemas de alarmas, todos estos ejemplos requieren de la integración de hardware y software embebido, todos ellos son el resultado de la Ingeniería Computacional.

Ciencias de la Computación (Computer Science)

Ciencias de la Computación abarca un amplio rango de fundamentos teóricos y de algoritmos en teoría de grafos utilizados en desarrollos en robótica, en visión por computador, en sistemas inteligentes, en bioinformática, entre otras áreas de gran interés. El campo de acción de las Ciencias de la Computación se divide en las siguientes tres categorías:

- El diseño e implementación de software que requiere de grandes retos en programación, y en donde es posiblemente necesario generar nuevos métodos.
- Nuevos usos del computador. Progresos en Ciencias de la Computación en áreas como las redes de datos, bases de datos y la interfaz humano-computador han permitido el desarrollo de Internet. Actualmente las Ciencias de la computación en conjunto con científicos de otros campos están trabajando en la construcción de robots que tengan conductas inteligentes, en el uso de bases de datos para crear nuevo conocimiento, y en el uso del computador para ayudar a descifrar los secretos de nuestro ADN.
- El desarrollo de formas eficientes para resolver problemas computacionales. Por ejemplo, las Ciencias Computacionales se preocupan por la mejor forma posible de almacenar la información en bases de datos, la mejor forma posible de enviar datos sobre las redes de datos, la mejor forma posible de mostrar imágenes complejas. La fundamentación teórica le permite a las Ciencias de la Computación trabajar en la determinación del mejor desempeño posible, y el estudio de algoritmos ayuda a desarrollar nuevos métodos que proveen un mejor desempeño.

Las Ciencias de la Computación abarcan un amplio rango de la teoría a través de la programación. El currículo de Ciencias de la Computación es muy amplio y es algunas veces criticado por la falta de preparación de sus graduados en trabajos específicos. Mientras otras disciplinas de la Ingeniería de Sistemas pueden producir graduados con más destreza específica en trabajos relacionados, las Ciencias de la Computación ofrecen una fundamentación detallada que permite a sus graduados la fácil adaptación a nuevas tecnologías y a nuevas ideas.

Sistemas de Información (Information Systems)

Los profesionales en Sistemas de Información se enfocan en soluciones integrales utilizando tecnología de la información y procesos de negocio para encontrar la información necesaria del negocio y de otras empresas, permitiendo alcanzar sus objetivos de una forma efectiva y eficiente. Esta disciplina tiene perspectivas en tecnologías de la información enfatizando en la información, y en la visión tecnológica con un instrumento para la generación, procesamiento y distribución de la información. Los profesionales

en esta disciplina están principalmente ocupados con la información que los sistemas computacionales pueden proveer para ayudar a las empresas en la definición y consecución de sus metas, y los procesos que una empresa puede implementar o mejorar usando tecnologías de la información. Ellos deben comprender tanto los factores técnicos como los factores empresariales, también deben poder ayudar a la empresa a determinar cual información y cual tecnología brindan ventajas competitivas.

Tecnología de la Información (Information Technology)

La Tecnología de la Información tiene dos grandes significados. El primero, en un sentido amplio el término Tecnología de la Información es a menudo utilizado para referirse a toda la computación. El segundo, en un sentido académico, se refiere a todos los programas de pregrado que preparan estudiantes en la tecnología computacional necesaria para los negocios, el gobierno, el cuidado hospitalario, los colegios y otra clase de organizaciones.

Los programas académicos de Tecnología de la Información existen para producir profesionales que posean una combinación adecuada de conocimiento y práctica, con destreza en el manejo y asistencia en la infraestructura tecnológica de la información de la organización y de las personas que usan ésta. Los profesionales de Tecnología de la Información asumen la responsabilidad de seleccionar productos de hardware y software apropiados para la organización, integrar estos productos con las necesidades e infraestructura de la organización, la instalación, configuración y mantenimiento de estas aplicaciones para los usuarios de cómputo de la organización.

Ejemplos de las responsabilidades de los profesionales de Tecnología de la Información, incluyen la instalación de redes, administración y seguridad de redes, el diseño de páginas web, el desarrollo de aplicaciones multimedia, la instalación de componentes de comunicación, el respaldo de sistemas de correos y la planificación y mantenimiento del ciclo de vida de la tecnología desde el punto de vista del mantenimiento, actualización y reemplazo.

Ingeniería del Software (Software Engineering)

La Ingeniería del Software es la disciplina que se dedica al desarrollo y mantenimiento de sistemas de software que se comportan de forma fidedigna y eficiente, con economía en el desarrollo y mantenimiento que satisfacen todos los requerimientos que los clientes han definido para ellos.

Los programas académicos de Ingeniería de Sistemas en las disciplinas de Ciencias de la Computación e Ingeniería del Software tienen muchos cursos en común. Los estudiantes de Ingeniería del Software aprenden más a cerca de confiabilidad y mantenimiento del software donde se enfocan principalmente en las técnicas para el desarrollo y mantenimiento del software desde el comienzo de su construcción. Mientras los estudiantes de Ciencias de la Computación, probablemente han escuchado de la importancia de tales

técnicas.

Los estudiantes de Ingeniería del Software aprenden como acceder a las necesidades del cliente y a desarrollar software para satisfacerlas.

2.1.3. Representación gráfica de los tópicos de las disciplinas de Ingeniería de Sistemas

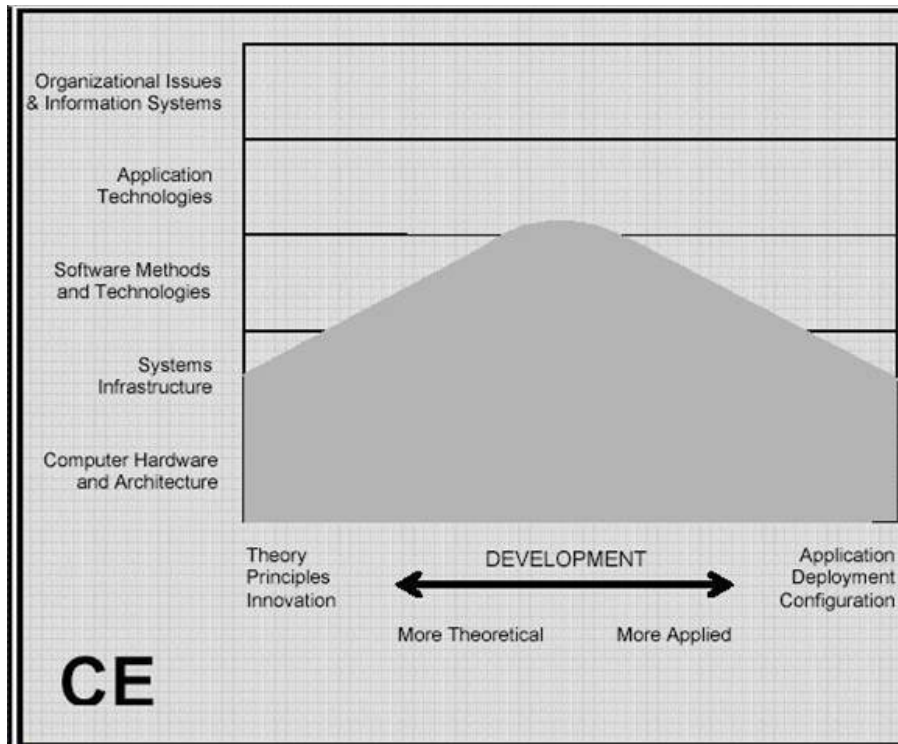
Para ilustrar las diferencias y semejanzas de las disciplinas de la Ingeniería de Sistemas, se presenta una caracterización gráfica de ellas. Las imágenes¹ sugieren como cada disciplina de la Ingeniería de Sistemas se ocupa del espacio de problemas computacionales actualmente, no de como debería de ser en el futuro, adicionalmente, la representación gráfica se enfoca principalmente en las áreas de desempeño de los egresados, no de todos los tópicos que un estudiante podría estudiar.

El rango horizontal va desde la teoría, principios e innovación en la izquierda, hasta aplicaciones, desarrollos y configuraciones en la derecha. Si alguien quiere trabajar en un laboratorio inventando nuevas cosas o en una universidad desarrollando nuevos principios, entonces debe estudiar una de las disciplinas de la Ingeniería de Sistemas que ocupe el espacio izquierdo. De forma contraria, si alguien quiere ayudar a las personas a elegir un uso apropiado de tecnología o quiere integrar productos disponibles para resolver problemas de las organizaciones, entonces debe estudiar una disciplina de Ingeniería de Sistemas que ocupe el espacio derecho. No se debe analizar únicamente los extremos derecho e izquierdo, también se debe considerar las posibilidades entre dichos extremos.

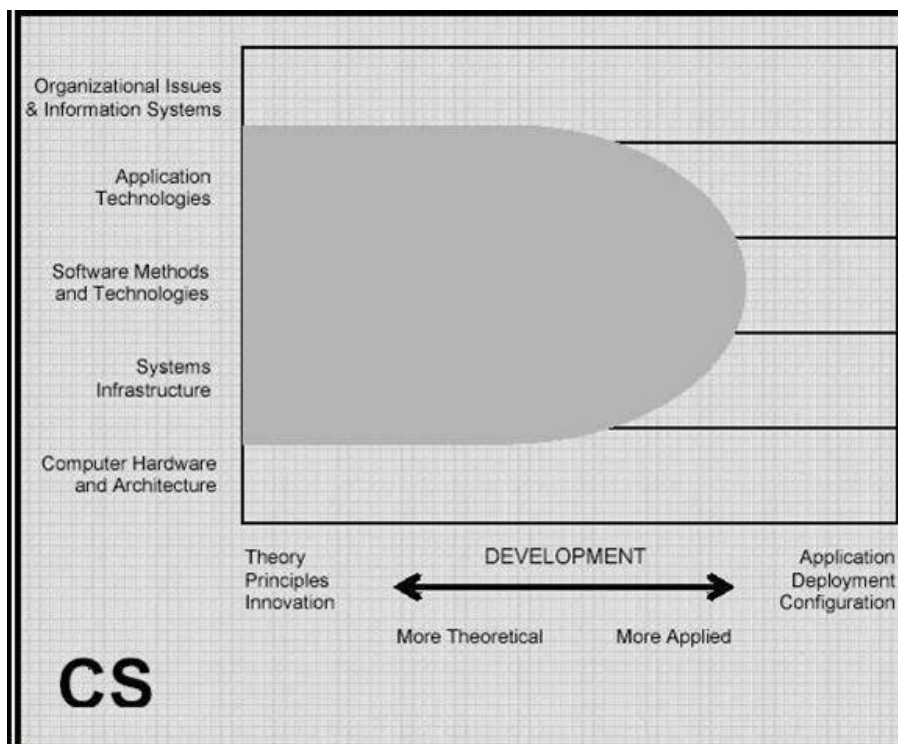
El rango vertical comienza en el fondo con “hardware y arquitectura del computador”, sigue con “infraestructura de sistemas”, luego “metodologías y tecnologías del software”, después con “aplicaciones tecnológicas”, hasta “sistemas de información y sus usos empresariales” en la parte superior. Al subir en el eje, la disciplina de la Ingeniería de Sistemas se enfoca en las personas, en la información y en la organización del espacio de trabajo. Al descender en el eje, la disciplina de la Ingeniería de Sistemas se enfoca en los dispositivos y la transmisión de datos entre ellos.

¹Todas las imágenes de esta subsección han sido tomadas de [ACMIEEE2005].

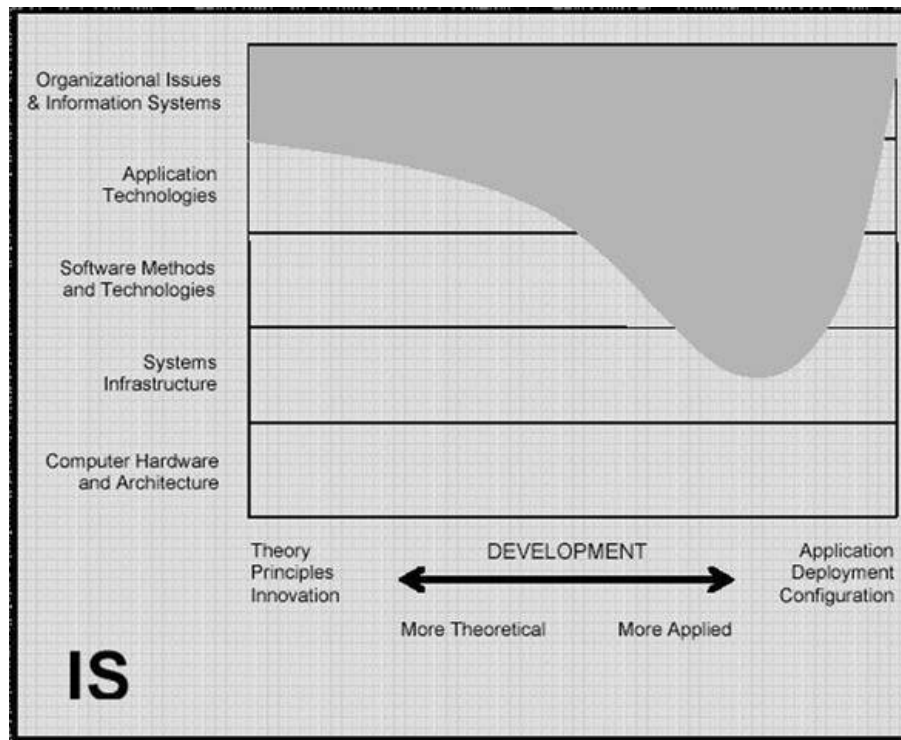
Tópicos de Ingeniería Computacional



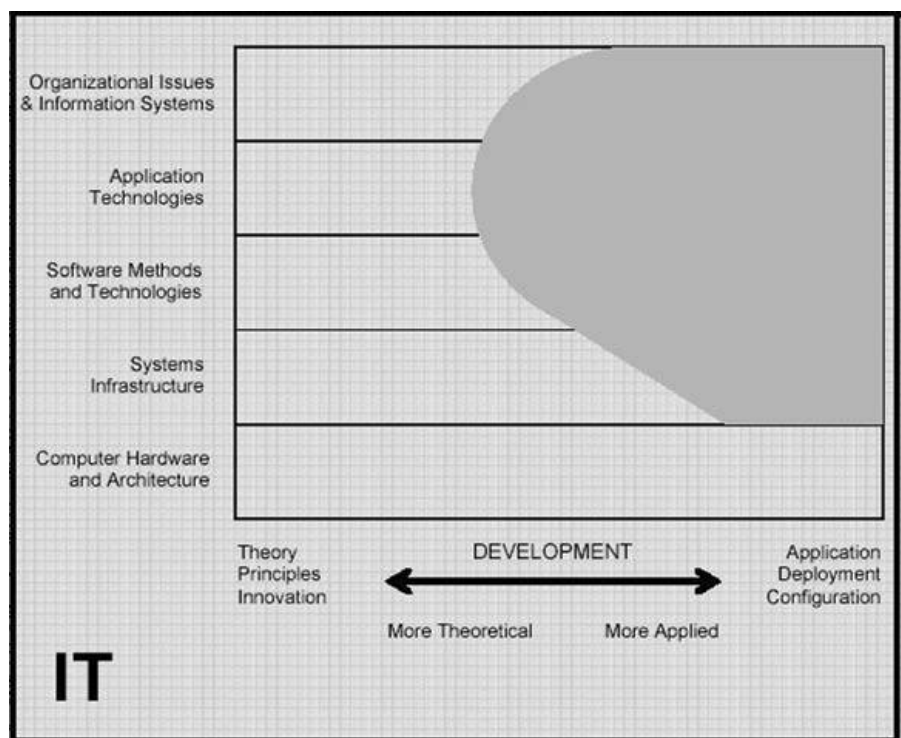
Tópicos de Ciencias de la Computación



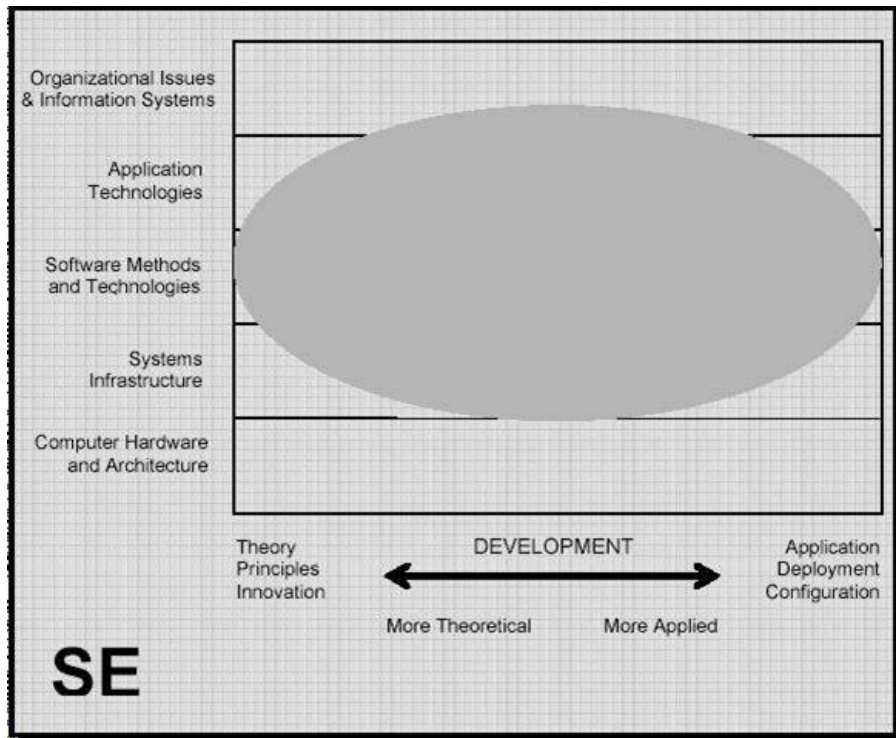
Tópicos de Sistemas de Información



Tópicos de Tecnología de la Información



Tópicos de Ingeniería del Software



2.2. Libros de amplia circulación en matemáticas computacionales con respecto al material de apoyo

Los libros que se revisaron con respecto a su material de apoyo para la enseñanza son:

- Estructuras de Matemáticas Discretas para la Computación de Bernard Kolman, Robert Busby y Sharon Ross
- Matemática Discreta de Norman Biggs
- Matemáticas Discreta y Combinatoria de Ralph Grimaldi
- Matemáticas Discretas de Kenneth Ross y Charles Wright
- Matemáticas Discretas de Richard Johnsonbaugh
- Matemáticas Discretas y sus Aplicaciones de Kenneth Rosen
- Matemáticas para la Computación de José Alfredo Jiménez Murillo

La investigación para cada uno de estos libros con respecto al material de apoyo para la enseñanza se presenta a continuación en las subsecciones siguientes.

2.2.1. Estructuras de Matemáticas Discretas para la Computación - Kolman, Busby y Ross

Este libro se encuentra en la sexta edición publicada en el año 2008 y es referenciado en la bibliografía como [KBR2008]. El libro no brinda ninguna clase de material de apoyo para la enseñanza.

2.2.2. Matemática Discreta - Norman L. Biggs

Este libro se encuentra en la segunda edición publicado en el año 2002 y es referenciado en la bibliografía como [B2002]. El material de apoyo que se encuentra en la página oficial del libro² es el siguiente:

- Capítulos uno y ocho de muestra
- Soluciones a los ejercicios propuestos en cada capítulos
- Lecturas de cada uno de los capítulos

2.2.3. Matemáticas Discreta y Combinatoria - Ralph Grimaldi

Este libro se encuentra en la quinta edición publicada en el año 2003 y es referenciado en la bibliografía como [G2003]. No se encontró ninguna clase de material de apoyo para la enseñanza.

2.2.4. Matemáticas Discretas - Ross y Wright

Este libro se encuentra en la quinta edición publicada en el año 2002 y es referenciado en la bibliografía como [RW2002]. No se encontró ninguna clase de material de apoyo.

2.2.5. Matemáticas Discretas - Richard Johnsonbaugh

Este libro se encuentra en la séptima edición publicada en el año 2008 y es referenciado en la bibliografía como [J2008]. Es muy pobre el material de apoyo que acompaña a este libro, únicamente en la página oficial del libro³ se encuentra:

- Enlaces de páginas de internet de otros autores que trabajan los temas
- Implementación en lenguaje C de algunos de los algoritmos trabajados en el libro

²<http://www.oup.co.uk/academic/companion/mathematics/discretemath/>

³<http://condor.depaul.edu/~rjohnson/dm7th/>

2.2.6. Matemáticas Discretas y sus Aplicaciones - Kenneth Rosen

El libro de Matemáticas Discretas y sus Aplicaciones (sexta edición, año 2007, referenciado en la bibliografía como [R2007]) de Kenneth Rosen brinda una buena cantidad de material de apoyo para los estudiantes, todo el material puede ser libremente descargado desde la página oficial del libro en la editorial Mc Graw Hill⁴. El material es el siguiente:

- Aplets de demostración interactiva (Interactive Demonstration Applets)
- Preguntas de Autoevaluación (Self Assessments)
- Aplicaciones de Matemáticas Discretas (Applications of Discrete Mathematics)
- Exploración de Matemáticas Discretas usando Maple (Exploring Discrete Math Using Maple)
- Una guía para escribir demostraciones matemáticas (A Guide to Writing Proofs)
- Errores comunes en Matemáticas Discretas (Common Mistakes in Discrete Math)
- Asesoramiento en la escritura de proyectos (Advice on the Writing Projects)
- Una guía de solución de ejemplos para estudiantes (Student's Solutions Guide sample)
- Ejemplos extra
- Diapositivas de clase

2.2.7. Matemáticas para la Computación - José A. Jiménez Murillo

El libro de Matemáticas para la Computación (primera edición, año 2009, referenciado en la bibliografía como [J2009]) de José Alfredo Jiménez Murillo brinda una buena cantidad de material de apoyo. El material es el siguiente:

- Aplicativo para realizar conversiones entre diferentes bases numéricas
- Preguntas de Autoevaluación
- Lecturas complementarias de otros autores
- Diapositivas complementarias de otros autores
- Enlaces de interés a páginas de otros autores que trabajan los temas
- Recursos multimedia de otros autores presentando los temas del curso
- Solución a los ejercicios propuestos por capítulo

⁴http://higherred.mcgraw-hill.com/sites/0072880082/student_view0/

2.3. Universidades extranjeras con material de apoyo en matemáticas computacionales

Esta sección nunca quedará completa, porque humanamente es imposible visitar todas las universidades del mundo para revisar si tienen material de apoyo en matemáticas computacionales, por este motivo se seleccionaron solo algunas universidades en el ámbito mundial.

2.3.1. MIT - Massachusetts Institute of Technology - Estados Unidos

A nivel mundial es posiblemente la universidad con mejor y mayor cantidad de material de apoyo para la enseñanza de los cursos.

Se revisaron los cursos de:

- Mathematics for Computer Science⁵ (Matemáticas para las Ciencias de la Computación)
- Theory of Computation⁶ (Teoría de la Computación)
- Automata, Computability, and Complexity⁷ (Autómatas, Computabilidad y Complejidad)
- Advanced Complexity Theory⁸ (Teoría de la Complejidad Avanzada)
- Introduction to Algorithms⁹ (Introducción a los Algoritmos, que realmente es un curso de Análisis de Algoritmos)

Para cada uno de estos curso se brinda material de:

- lecturas
- asignaciones
- exámenes

Es normal que también se brinden los videos de las clases de cada uno de los temas, con una ventaja adicional, en los videos se han grabado dictando clase en el MIT a los autores o coautores de los libros de mayor utilización a nivel mundial en cada uno de

⁵<http://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-042j-mathematics-for-computer-science-spring-2005/>

⁶<http://ocw.mit.edu/courses/mathematics/18-404j-theory-of-computation-fall-2006/>

⁷<http://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-045j-automata-computability-and-complexity-spring-2005/>

⁸<http://ocw.mit.edu/courses/mathematics/18-405j-advanced-complexity-theory-fall-2001/>

⁹<http://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-046j-introduction-to-algorithms-sma-5503-fall-2005/>

estos temas.

El MIT ha incentivado a sus docentes para que escriban muy buen material de apoyo para la enseñanza de sus cursos, material que a la postre se ha convertido en libros de muy buena calidad en estos temas y de gran utilización a nivel mundial, este es el caso del libro “Introduction to the Theory of Computation” del profesor Michael Sipser (libro referenciado en la bibliografía como [S1996]), quien es el profesor titular del curso de “Teoría de la Computación”, donde obviamente el texto guía del curso es su libro. Este mismo caso también se presenta con el curso de “Introducción a los Algoritmos” dictado por el profesor Charles E. Leiserson, donde se utiliza el libro “Introduction to Algorithms” del cual es coautor (libro referenciado en la bibliografía como [CLRS2001]), este libro fue editado en conjunto por MIT Press y Mc Graw Hill.

2.3.2. Princeton University - Estados Unidos

En la Universidad de Princeton se revisó el curso de “Algoritmos y Estructuras de Datos” (Algorithms and Data Structures¹⁰) de los profesores Robert Sedgewick y Kevin Wayne.

Para el curso se brinda material de:

- lecturas
- diapositivas de clase
- asignaciones
- demos
- exámenes
- solución de los exámenes

La Universidad de Princeton también le ha apostado a que sus docentes escriban muy buenos libros y que alrededor de ellos también se genere muy buen material de apoyo para su enseñanza. Los textos guías de este curso son “Algorithms”, “Algorithms in Java” de Robert Sedgewick (referenciados en la bibliografía como [S1983], [S2002a] y [S2002b] respectivamente).

2.3.3. Universidad de Málaga - España

En la Universidad de Málaga se revisó el curso de “Matemática Discreta”¹¹ de los profesores Francisco J. Rodríguez y Pablo J. Cordero.

Para el curso se brinda material de:

¹⁰<http://www.cs.princeton.edu/courses/archive/spring08/cos226/info.html>

¹¹<http://sevein.matap.uma.es/~aciego/mat-es-2005.html>

- diapositivas de clase
- talleres

Los profesores presentan unas muy buenas diapositivas de clase construidas con el lenguaje de programación por etiquetas \LaTeX .

En la Universidad de Málaga también se revisó la página del profesor Antonio Vallecillo, desde donde se puede descargar libremente el libro “Técnicas de Diseño de Algoritmos”¹² (libro referenciado en la bibliografía como [GV1998]), el cual es utilizado como material de apoyo del curso “Análisis y Diseño de Algoritmos” en ésta universidad.

2.3.4. Universidad Simón Bolívar - Venezuela

En la Universidad Simón Bolívar se revisaron las asignaturas de “Estructuras Discretas I, II y III”¹³ del profesor Vicente Yriarte.

Para las asignaturas se brinda material de:

- ejercicios
- tareas
- exámenes
- libros
- diapositivas de clase

El profesor Vicente Yriarte ha escrito los libros de “Elementos de Teoría Combinatoria” (Septiembre de 1996, referenciado como [Y1996]) y “Elementos de Teoría Axio-mática de Conjuntos (Octubre de 2001, referenciado como [Y2001]) como material de apoyo de sus asignaturas de “Estructuras Discretas”. Los libros pueden ser libremente descargados desde la página oficial del profesor, estos han sido escritos utilizando el lenguaje de programación por etiquetas \LaTeX .

2.3.5. Instituto Tecnológico de Nuevo Laredo - Mexico

En el Instituto Tecnológico de Nuevo Laredo se revisó la página oficial del Ingeniero Bruno López Takeyas¹⁴, donde se encontró material de apoyo para la enseñanza de Matemáticas Discretas, Teoría de Grafos, Estructuras de Datos e Inteligencia Artificial.

Para las asignaturas se brinda material de:

- ejercicios
- diapositivas de clase

¹²<http://www.lcc.uma.es/~av/Libro/>

¹³<http://www.ldc.usb.ve/~yriarte/>

¹⁴<http://www.itnuevolaredo.edu.mx/takeyas>

2.4. Universidades colombianas con material de apoyo en matemáticas computacionales

Según las estadísticas del Examen de Calidad de la Educación Superior del 2009, en Colombia existen al rededor de 142 programas de Ingeniería de Sistemas o afines, de éste total de programas, en la investigación del material de apoyo se le dio prioridad a programas de Ingeniería de Sistemas que pertenecen a universidades acreditadas de alta calidad según el Ministerio de Educación Nacional.

2.4.1. Universidad de los Andes - Bogotá

En la Universidad de los Andes se revisó la página oficial del matemático e ingeniero de sistemas Rodrigo Cardoso¹⁵, donde se encontró material de apoyo para la enseñanza de las asignaturas:

- Fundamentos Matemáticos para Informática
- Teoría de Lenguajes
- Eficiencia de Algoritmos
- Matemática Informática

Para las asignaturas se brinda material de:

- parciales y soluciones
- tareas y soluciones
- diapositivas de clase
- enunciados de proyectos de fin de asignatura
- libros

La Universidad de los Andes ha promovido entre sus docentes una política de escritura de libros en el área de su desempeño, se han producido libros que son unos verdaderos clásicos en Ingeniería de Sistemas en el contexto colombiano, como el libro de “Elementos de Lógica Matemática y Calculabilidad” del profesor Xavier Caicedo (libro referenciado en la bibliografía como [C1989]).

El profesor Rodrigo Cardoso ha escrito el libro “Verificación y Desarrollo de Programas” (libro referenciado en la bibliografía como [C1993]), el cual es utilizado como material de apoyo en algunos temas de sus diferentes asignaturas. Éste libro puede ser libremente descargado desde la página oficial del autor en la Universidad de los Andes.

¹⁵<http://agamenon.uniandes.edu.co/~rcardoso/>

2.4.2. Universidad EAFIT - Medellín

En la Universidad EAFIT se revisó la página oficial del profesor Andrés Sicard-Ramírez¹⁶, quien es el profesor líder de las asignaturas:

- Lógica Combinatoria
- Teoría de Grafos
- Teoría de Lenguajes
- Lógica

El profesor Andrés Sicard-Ramírez ha producido material en conjunto con el profesor Raúl Gómez Marín, donde se le ha dado prioridad a la escritura del libro “Informática Teórica. Elementos Propedéuticos”, editado por la Universidad EAFIT (libro referenciado en la bibliografía como [GS2002]), libro utilizado como texto guía de la asignatura “Teoría de Lenguajes”. Éste libro puede ser libremente descargado desde la página oficial del profesor Andrés Sicard-Ramírez en la Universidad EAFIT. Adicionalmente este par de profesores han generado un buen material de apuntes de clase para la asignatura de “Teoría de Grafos”, tanto el libro como los apuntes de clase han sido escritos utilizando el lenguaje de programación por etiquetas \LaTeX .

Desde la página oficial del profesor Andrés Sicard-Ramírez se puede descargar libremente el libro “Argumentación Deductiva con Diagramas y Árboles de Forzamiento”¹⁷ del profesor Manuel Sierra Aristizábal (libro referenciado en la bibliografía como [S2007]), éste libro es el texto guía del curso de “Lógica” orientado por el profesor Andrés Sicard.

2.4.3. Universidad ICESI - Cali

En la Universidad ICESI únicamente se puede revisar los exámenes realizados y los programas de las asignaturas. Las asignaturas revisadas fueron las siguientes:

- Algebra y Funciones
- Lógica y Argumentación
- Matemáticas Discretas
- Informática Teórica
- Algoritmos y Estructuras de Datos

En la construcción de los exámenes revisados han participado principalmente los profesores:

¹⁶<http://www1.eafit.edu.co/asicard/>

¹⁷<http://www1.eafit.edu.co/asicard/teaching/logic-CB0260/SierraAristizabal-2007.Argumentacion-Deductiva-con-Diagramas-y-Arboles-de-Forzamiento.pdf.gz>

- Aníbal Sosa
- Carlos Ernesto Ramírez
- Alfonso Bustamante Arias

La asignatura de “Lógica y Argumentación” pertenece a todas las carreras de pregrado ofrecidas en la Universidad ICESI, se utiliza como texto guía el libro “Lógica y Argumentación: De los argumentos inductivos a las álgebras de Boole” del profesor del ICESI, Alfonso Bustamante¹⁸ (libro referenciado en la bibliografía como [B2009]). Otro libro del profesor Alfonso Bustamante, el de “Elementos de Algebra en Ciencias de la Computación” es material de referencia del curso de “Matemáticas Discretas” (libro referenciado en la bibliografía como [B1988]).

2.4.4. Universidad del Valle - Cali

En la Universidad del Valle se revisaron las siguientes asignaturas¹⁹:

- Matemáticas Discretas I
- Matemáticas Discretas II
- Fundamentos de Lenguajes de Programación
- Fundamentos de Análisis y Diseño de Algoritmos

Para cada una de éstas asignaturas se encuentra como material de apoyo, unas buenas diapositivas de clase para cada uno de los temas.

Los profesores que han participado en la construcción de estas diapositivas de clase son:

- Juan Francisco Díaz F.
- Julio César López
- Raúl Ernesto Gutierrez de Piñerez
- Orlando Arboleda Molina.
- James Yerson Ortiz V.

¹⁸http://www.icesi.edu.co/es/programas/pregrado/sistemas/profesores/alfonso_bustamante.htm

¹⁹<http://eisc.univalle.edu.co/materias/>

2.4.5. Universidad Nacional - Bogotá

En la Universidad Nacional se revisó la página oficial del profesor Rodrigo de Castro Korgi²⁰, página obsoleta que no se actualiza desde 1999 y que no contiene ningún tipo de material de apoyo.

El profesor Rodrigo de Castro ha escrito el libro “Teoría de la Computación: lenguajes, autómatas, gramáticas”, editado por el Departamento de Matemáticas de la Universidad Nacional de Colombia (libro referenciado en la bibliografía como [dC2004]), éste libro es el actual texto guía de la asignatura “Teoría de la Computación” en la Universidad Nacional de Colombia, y de la asignatura “Informática Teórica”²¹ de la Universidad ICESI de Cali.

El profesor Rodrigo de Castro tiene otro libro muy famoso entre los usuarios del lenguaje de programación por etiquetas \LaTeX , llamado “El Universo \LaTeX ” (libro referenciado en la bibliografía como [dC2003]), éste libro es posiblemente uno de los mejores libros a nivel mundial escritos en lengua española sobre este tema.

De nuevo, las universidades más prestigiosas a nivel mundial y nacional le apuestan a que sus docentes escriban muy buenos libros en su área de desempeño, apostándole a que sus libros se vuelvan unos verdaderos clásicos en el tema y que adicionalmente sean una referencia obligada. Con la escritura de libros las universidades ganan en prestigio y reconocimiento, y van marcando la pauta del qué y del cómo debería enseñarse determinados temas.

2.4.6. Pontificia Universidad Javeriana - Cali

En la Pontificia Universidad Javeriana de Cali, se revisó la página oficial de la asignatura “Análisis y Diseño de Algoritmos”²² orientada por el ingeniero de sistemas Andrés Becerra Sandoval, donde se encontró el siguiente material de apoyo para la enseñanza de la asignatura:

- tareas
- diapositivas de clase
- libros

Es valioso el material de apoyo que presenta el Ing. Andrés Becerra, pero lo es aún más, las referencias bibliográficas de los libros:

- “Algorithms”²³ de S. Dasgupta, C. H. Papadimitriou, y U. V. Vazirani (referenciado en la bibliografía como [DPV2006]). Libro que se puede descargar directa-

²⁰<http://www.matematicas.unal.edu.co/rdcastro/>

²¹http://www.icesi.edu.co/portal/pls/portal/psiaepre.pprecon_contactual?pdescripmat_codigo=08289

²²http://cic.puj.edu.co/wiki/doku.php?id=materias:analisis_y_diseño_de_algoritmos

²³<http://www.cs.berkeley.edu/~vazirani/algorithms.html>

mente desde la página oficial del profesor Vazirani en la Universidad de Berkeley (Estados Unidos).

- “Técnicas de Diseño de Algoritmos”²⁴ de Rosa Guerequeta y Antonio Vallecillo (referenciado en la bibliografía como [GV1998]). Libro que se puede descargar directamente desde la página oficial del profesor Antonio Vallecillo en la Universidad de Málaga (España).

En la Pontificia Universidad Javeriana de Cali, también se revisó la página oficial de la asignatura “Computabilidad y Lenguajes Formales”²⁵ orientada por la ingeniera de sistemas Gloria Inés Álvarez, donde se encontró el siguiente material de apoyo para la enseñanza de la asignatura:

- tareas individuales
- diapositivas de clase
- software JFLAP²⁶, el cual puede ser libremente descargado desde la página oficial del proyecto.

Los ingenieros Gloria Inés Álvarez y Andrés Becerra han escrito sus diapositivas de clase utilizando el lenguaje de programación por etiquetas \LaTeX .

2.4.7. Universidad de Antioquia - Medellín

En la Universidad de Antioquia únicamente se puede revisar los contenidos de las asignaturas. Las asignaturas revisadas fueron las siguientes:

- Lógica y Representación I (primer semestre)
- Matemáticas Discretas I (primer semestre)
- Lógica y Representación II (segundo semestre)
- Matemáticas Discretas II (segundo semestre)
- Lógica y Representación III (tercer semestre)
- Teoría de Lenguajes y Laboratorio (cuarto semestre)

²⁴<http://www.lcc.uma.es/~av/Libro/>

²⁵<http://cic.puj.edu.co/wiki/doku.php?id=materias:computabilidad>

²⁶<http://www.cs.duke.edu/csed/jflap/>

2.4.8. Universidad del Norte - Barranquilla

En la Universidad del Norte únicamente se pueden revisar los contenidos de las asignaturas. Las asignaturas revisadas fueron las siguientes:

- Matemáticas Discretas (cuarto semestre)
- Estructuras Discretas (quinto semestre)
- Algoritmos y Complejidad (quinto semestre)
- Compiladores (octavo semestre)

2.5. Contenido de matemáticas computacionales según el Ministerio de Educación Nacional

Los temas de Matemáticas Computacionales que actualmente sugiere el Ministerio de Educación Nacional para un programa de Ingeniería de Sistemas, han sido tomados del documento de Acofi-Icfes del 2003 sobre Especificaciones de los Exámenes de Estado de Calidad de la Educación en Ingeniería de Sistemas/Informática [AI2003]. Los temas de Matemáticas Computacionales se encuentran en las siguientes áreas :

Matemáticas discretas:

- Lógica:
 - Lógica proposicional, conectivos lógicos, tablas de verdad.
 - Forma normal conjuntiva / disyuntiva.
 - Fórmulas válidas / tautologías.
 - Lógica de predicados, cuantificación universal / existencial.
 - Límites de la lógica.
 - Técnicas de demostración:
 - Estructura de pruebas formales.
 - Prueba directa, por contradicción, por contraposición.
 - Inducción matemática.
 - Ordenes bien fundados.
 - Definiciones matemáticas recursivas.
- Conteo, ecuaciones de recurrencia:
 - Reglas de suma y producto.
 - Recursión.
 - Progresiones aritméticas y geométricas.

- Principio del palomar.
- Permutaciones y combinaciones.
- Ecuaciones de recurrencia lineales.
- Funciones, relaciones, conjuntos:
 - Conjuntos (operaciones básicas, producto cartesiano, potencia, diagramas de Venn).
 - Relaciones binarias (propiedades: reflexividad, simetría, transitividad, equivalencia, clausuras, inversas, composición).
 - Funciones (como relaciones unívocas; propiedades: inyección, sobreyección, biyección).
- Grafos:
 - Árboles.
 - Grafos dirigidos, no dirigidos.
 - Recorrido de grafos.
 - Recursión.

Programación y algorítmica:

- Estructuras de datos:
 - Tipos primitivos.
 - Arreglos, registros, cadenas.
 - Representación de datos en memoria.
 - Manejo dinámico de almacenamiento.
 - Apuntadores y referencias.
 - Estructuras dinámicas (pilas, colas, árboles, grafos, entre otros).
 - Recursión (en datos).
 - Archivos.
 - Estrategias para elegir la estructura de datos correcta.
- Análisis de algoritmos:
 - Complejidad
 - Algoritmos polinomiales / exponenciales
- Algoritmos clásicos:

- Búsqueda
- Ordenamiento
- Ruta mínima en grafos
- Verificación de programas:
 - Especificación.
 - Corrección.
 - Aserciones.
 - Invariantes de ciclo.
- Límites de la algorítmica:
 - Indecidibilidad.
 - Clases P y NP.

Informática teórica:

- Lenguajes Formales y Autómatas:
 - Conceptos básicos de lenguajes formales
 - Definición
 - Gramáticas / BNF
 - Máquinas de Turing
 - Definición, fundamentos.
 - Autómatas
 - Estados, transiciones
 - Autómatas determinísticos, no determinísticos
 - Modelaje con autómatas
 - Autómatas finitos / Relación con lenguajes regulares
 - Autómatas de pila / Relación con lenguajes libres de contexto
 - Semántica de lenguajes
 - Herramientas de análisis sintáctico
 - Cómo se utilizan
 - Limitaciones

2.6. Exámenes de calidad en la educación superior

Los Exámenes de Calidad de la Educación Superior (ECAES) son para el Ministerio de Educación un instrumento de la medida de la calidad, así como también lo es el registro calificado de cumplimiento de estándares mínimos de calidad, con una gran diferencia, los ECAES son realizados cada año y la visita de pares para registro calificado son realizadas cada siete años, antes del vencimiento de la vigencia del registro actual.

Por más que el Ministerio de Educación quiera negarlo, los resultados de los ECAES han generado un ranking entre lo que son supuestamente las “mejores” universidades o programas y las “peores”, esto tiene un trasfondo comercial que es bien utilizado por aquellas institución que salen bien calificadas y que es relegado al olvido (o más bien nunca mencionado) por aquellas instituciones que salen mal calificadas.

Aproximadamente en Octubre del 2007 salió el “boom” periodístico: “La Corte Constitucional tumba los ECAES”, noticia que lo que hizo realmente fue desinformar, porque al leer todo el párrafo de la noticia, lo que realmente se decía era que la Corte Constitucional emitió un fallo que declaró inexecutable el Artículo 8 de la Ley 749 de 2002, el cual respaldaba el Decreto 1781 de 2003, decreto por medio del cual fueron creado los ECAES. A pesar de esto, los ECAES siguieron vigentes en 2007 y 2008, pues la Corte estipuló como plazo el 16 de diciembre de 2008 para que el Congreso de la República emitiera una reglamentación legal sobre los ECAES, esta reglamentación fue expedida en la Ley 1324 del 13 de Julio de 2009. En el artículo 7 de la presente ley se establece la obligatoriedad del ECAES para los estudiantes de pregrado de cualquier institución que imparta educación superior.

El gobierno colombiano para dar cumplimiento a la Ley 1324 emana el Decreto 3963 del 14 de Octubre de 2009, decreto en el que en el parágrafo del artículo 8 se establece que el ECAES es un correquisito de grado, adicionalmente en el artículo 7 se establece que para los estudiantes de programas académicos que al 2008 no tenían ECAES, estos serán evaluado con pruebas de competencias genéricas.

La evaluación de la calidad no es solo un instrumento de medición de la calidad, es un proyecto que en el año 2009 le generó al gobierno colombiano mucho más de doce mil doscientos cuarenta millones de pesos (12.240'000.000). En el año 2009 se inscribieron para presentar el ECAES (entre estudiantes y egresados) doscientas cuarenta mil personas (240.000), el valor del ECAES para estudiantes de universidades publicas es de cincuenta y un mil pesos (51.000) y para estudiantes de universidades privadas y egresados es de sesenta y seis mil pesos (66.000). Si se considera que todos los evaluados pagan el ECAES en la categoría más barata entonces:

$$240,000 * 51,000 = 12,240'000,000$$

pero obviamente no todos los evaluados están en la categoría más económica, por este motivo el dinero recaudado por el gobierno es mucho mayor, lo que se pretende es dar una cota inferior del dinero generado por los ECAES del año 2009. Los valores y el total

de evaluados de los ECAES 2009 han sido tomados de los Boletines de Prensa 82 y 95 del ICFES respectivamente.

Capítulo 3

Propuesta de asignaturas en matemáticas computacionales

El programa de Ingeniería de Sistemas y Computación de la Universidad Tecnológica de Pereira pertenece a la línea de Ciencias de la Computación que a nivel internacional se ha definido por ACM e IEEE en “Computing Curricula 2005” [ACMIEEE2005]. Para dar cumplimiento a éste estándar internacional en Ciencias de la Computación en el área de Matemáticas Computacionales, se proponen cuatro asignaturas, de las cuales se presentan sus programas en éste capítulo.

Al dar cumplimiento en el área de las Matemáticas Computacionales según el estándar internacional exigido para Ciencias de la Computación, también se da cumplimiento a la directriz que en este mismo tema ha dado el Ministerio de Educación Nacional para un programa de Ingeniería de Sistemas en Colombia [AI2003].

3.1. Programa de matemáticas computacionales I

El primer curso de matemáticas computacionales contiene los temas que tradicionalmente se trabajan en un curso de Matemáticas Discretas I, el temario de éste curso se desarrolla por completo en éste trabajo de grado de Maestría. El programa sugerido de la materia es el siguiente:

Propuesta de Programa para el PRIMER Curso de Matemáticas Computacionales En Programas de Estudio de Ingeniería de Sistemas

Semestre sugerido: Tercero

Créditos sugeridos: 3

Intensidad horaria sugerida: 4 horas semanales

Requisitos sugeridos: Matemáticas II y Programación I

Propuesta realizada por: Ing. Hugo Humberto Morales Peña

JUSTIFICACIÓN

La Programación, las Bases de Datos, el Análisis de Algoritmos y la Inteligencia Artificial, entre otros, se fundamentan en los temas trabajados en Matemáticas Discretas. Por consiguiente, es de vital importancia que los estudiantes de Ingeniería de Sistemas accedan a este conocimiento desde una perspectiva puramente formal.

OBJETIVOS

Objetivo General

Presentar algunas de las estructuras matemáticas discretas fundamentales para la computación, tales como, la lógica, la teoría de conjuntos y la noción formal de prueba y explicar su relación directa con áreas tales como Análisis de Algoritmos, Ingeniería del Software y Bases de Datos.

Objetivos Específicos

- Comprender los conceptos básicos de teoría de conjuntos, funciones, relaciones y sus operaciones asociadas.
- Aplicar estos conceptos en el modelamiento de problemas prácticos.
- Utilizar métodos formales en la lógica proposicional y en la de predicados.
- Usar la lógica simbólica para la especificación de problemas.
- Comprender y manejar la relación que existe entre la prueba por inducción matemática con los conceptos de recursión y de estructuras definidas recursivamente.
- Poder utilizar técnicas de pruebas para resolver problemas de otras áreas de la computación, tales como especificación y verificación formal, Ingeniería de Software y Bases de Datos.
- Utilizar métodos formales para la escritura y presentación de ideas.

METODOLOGÍA

Clases

El curso tendrá cuatro (4) horas de intensidad semanal separadas en dos bloques de estudio cada uno de dos horas.

Lecturas

Cada unidad del programa tendrá lecturas complementarias con el material de apoyo del curso, las lecturas le permitirán al estudiante obtener un panorama más amplio sobre el tema. La responsabilidad de estas lecturas corresponde directamente al estudiante.

Texto

Adicional al material de apoyo del curso, se tendrá como texto guía el libro: Discrete Mathematics and Its Applications, de Kenneth Rosen [R2007].

EVALUACIÓN

■ Primer examen parcial	25 %
■ Segundo examen parcial	25 %
■ Examen final	25 %
■ Tareas, talleres y quices	25 %

CONTENIDO DEL CURSO**Unidad 1 (4 semanas): Introducción a la Lógica Matemática**

- Lógica proposicional
- Leyes de la lógica proposicional
- Simplificación de expresiones proposicionales utilizando las leyes de la lógica
- Reglas de inferencia
- Lógica de predicados
- Cuantificador universal
- Cuantificador existencial
- Aplicaciones

Justificación: El tema de lógica es necesario en los estudiantes de Ingeniería de Sistemas y Computación porque brinda la fundamentación matemática necesaria para los cursos de Electrónica Digital en la parte de diseño de circuitos lógicos.

Unidad 2 (2.5 semanas): Uso de Sucesiones y Sumatorias en Computación

- Funciones especiales: piso y techo
- Sucesiones
- Sumatorias
- Las sumatorias dobles y triples y los ciclos de repetición en programación
- Definiciones recursivas

- Notaciones computacionales (O , Ω y Θ)
- Complejidad de algoritmos iterativos
- El mito del alto costo computacional de la programación recursiva

Justificación: El tema de sucesiones y sumatorias es de gran importancia para los estudiantes de Ingeniería de Sistemas y Computación porque brindan las bases para el análisis de la eficiencia computacional de los algoritmos iterativos.

Unidad 3 (2 semanas): Técnicas de Demostración

- Estructura de una demostración
- Demostración directa
- Demostración por contra ejemplo
- Demostración por contradicción
- Demostración por contra recíproca
- Demostración por casos
- Demostración por inducción matemática

Justificación: El tema de técnicas de demostración no es solo importante para los estudiantes de Ingeniería de Sistemas y Computación, en general es importante para todo tipo de estudiantes que en su carrera tenga diferentes asignaturas de matemáticas. Es importante que los estudiantes de Ingeniería de Sistemas tengan las competencias necesarias para demostrar o refutar que un argumento matemático se verdadero o falso para su posterior uso en soluciones computacionales.

Unidad 4 (2 semanas): Relaciones de Recurrencia

- Relaciones de recurrencia de primer grado
- Solución de relaciones de recurrencia de primer grado
- Utilización de la técnica de demostración por inducción matemática para ratificar o refutar que la solución de una relación de recurrencia es o no correcta.
- Complejidad de algoritmos recursivos.

Justificación: El tema de la solución de relaciones de recurrencia es de gran importancia para los estudiantes de Ingeniería de Sistemas y Computación porque son el tema clave para el cálculo de la complejidad de los algoritmos recursivos.

Unidad 5 (3 semanas): Conjuntos, Funciones, Relaciones y Bases de Datos Relacionales

- Conjuntos y operaciones de conjuntos
- Funciones (conceptos fundamentales)
- Función inyectiva, función sobreyectiva y función biyectiva
- Relaciones y sus propiedades
- Representación de relaciones
- Relaciones de equivalencia
- Clases de equivalencia y particiones
- Conjuntos parcialmente ordenados
- Introducción al modelo de bases de datos relacional
- Introducción al álgebra relacional
- Introducción al lenguaje de consulta estructurado (SQL)

Justificación: Los temas de conjuntos y funciones son la base matemática necesaria para los diferentes curso de programación que recibe en la formación los estudiantes de Ingeniería de Sistemas. De forma similar, los temas de conjuntos y relaciones son la base matemática sobre la cual se apoya el modelo de bases de datos relacional, el álgebra relacional y el lenguaje de consulta estructurado (SQL).

Unidad 6 (2.5 semanas): Introducción a la teoría de números

- El algoritmo de la división
- Los números primos y los números compuestos
- Teorema fundamental de la aritmética
- El Máximo Común Divisor (MCD)
- El Mínimo Común Múltiplo (MCD)
- El algoritmo de Euclides
- El teorema de la congruencia
- La aritmética modular y sus aplicaciones
- Representación de números en el computador
- Representación de enteros en diferentes bases

Justificación: Los temas que componen la teoría de números brindan a los estudiantes de Ingeniería de Sistemas y Computación la base matemática para comprender como los diferentes tipos de números son almacenados en el computador, como la información es almacenada en el computador y del gran uso que se le da al Teorema Fundamental de la Aritmética en la computación.

PARCELACIÓN DEL CURSO POR CLASES

1. Presentación del curso, forma y fechas de evaluación. Introducción a la lógica matemática.
2. Introducción al cálculo proposicional.
3. Reglas del cálculo proposicional. Simplificación de expresiones del cálculo proposicional por medio de la utilización de las reglas del cálculo proposicional
4. Conectivos NAND (\uparrow), NOR (\downarrow) y X-OR (\otimes) en el cálculo proposicional. Entrega a los estudiantes del primer taller calificable por grupos.
5. Reglas de inferencia.
6. Utilización de las reglas de inferencia para verificar si un argumento es válido.
7. Introducción al cálculo de predicados. Entrega a los estudiantes del segundo taller calificable por grupos.
8. Introducción al cálculo de predicados (continuación).
9. Funciones especiales: piso y techo. Sucesiones.
10. Recursividad (funciones recursivas). El mito del alto costo computacional de la programación recursiva.
11. Sumatorias. Sumatorias dobles y triples y los ciclos anidados.
12. Notaciones computacionales y complejidad de algoritmos iterativos.
13. Primera evaluación parcial.
14. Técnicas de demostración: directa.
15. Técnicas de demostración: por contra ejemplo, contra-recíproca y por contradicción.
16. Técnica de demostración por casos.
17. Inducción matemática. Entrega a los estudiantes del tercer taller calificable por grupos.
18. Relaciones de recurrencia.

19. Solución de relaciones de recurrencia de primer grado. Entrega a los estudiantes del cuarto taller calificable por grupos.
20. Solución de ejemplos de relaciones de recurrencia donde primero se halle la serie equivalente a la relación de recurrencia, luego se solucione la serie y por ultimo se demuestre por medio de inducción matemática que la solución a la relación de recurrencia quedo o no correcta.
21. Complejidad de algoritmos recursivos.
22. Segunda evaluación parcial.
23. Conjuntos y operaciones de conjuntos. Funciones. Conceptos fundamentales, propiedades: inyectiva, sobreyectiva y biyectiva. Representación gráfica.
24. Relaciones. Conceptos fundamentales, propiedades y representación gráfica.
25. Relaciones de equivalencia. Conjuntos parcialmente ordenados. Entrega a los estudiantes del quinto taller.
26. Introducción al modelo de bases de datos relacional y al álgebra relacional.
27. Introducción al lenguaje de consulta estructurado (SQL).
28. Números enteros, división de enteros, el algoritmo de la división, números primos y el teorema fundamental de la aritmética.
29. Definición de Máximo Común Divisor (MCD) y mínimo común múltiplo (MCM). El algoritmo de Euclides. Entrega a los estudiantes del sexto taller donde se propone la implementación en lenguaje C de los algoritmos de máximo común divisor, mínimo común múltiplo, números primos y Euclides.
30. Aritmética modular y sus aplicaciones: el sistema criptográfico de Julio Cesar.
31. Representación de números enteros en el computador, cambio de base de los números enteros.
32. Algoritmos para sumar y multiplicar números enteros escritos en base dos. Entrega a los estudiantes del séptimo taller donde se propone la implementación en lenguaje C de los algoritmos de cambio de base, suma y multiplicación en base dos y el encriptamiento de Julio Cesar.
33. Evaluación final.

BIBLIOGRAFÍA

- [B2002] | Biggs, Norman L.. 2002. “Discrete Mathematics”, Second Edition. Oxford University Press.
- [B1988] | Bustamente Arias, Alfonso. 1988. “Elementos de Algebra en Ciencias de la Computación”. Serie de textos universitarios, Universidad ICESI, Cali, Colombia.
- [B2009] | Bustamente Arias, Alfonso. 2009. “Lógica y Argumentación: De los argumentos inductivos a las álgebras de Boole”. Pearson Educación. Colombia.
- [G2003] | Grimaldi, Ralph. 2003. “Discrete and Combinatorial Mathematics”, Five Edition. Pearson.
- [J2008] | Johnsonbaugh, Richard. 2008. “Discrete Mathematics”. Seven Edition. Prentice Hall.
- [J2009] | Jiménez Murillo, José A.. 2009. “Matemáticas para la Computación”. Alfaomega.
- [KBR2008] | Kolman, Bernard, Busby, Robert y Ross, Sharon C.. 2008. “Estructuras de Matemáticas Discretas para la Computación”, Sexta edición. Prentice Hall.
- [R2007] | Rosen, Kenneth H.. 2007. “Discrete Mathematics and Its Applications”, Six edition. Mc Graw Hill.
- [RW2002] | Ross, Kenneth A. y Wright, Charles R.. 2002. “Matemáticas Discretas”, Quinta edición. Prentice Hall.
- [Y2001] | Yriarte, Vicente. 2001. “Elementos de Teoría Axiomática de Conjuntos”. Universidad Simón Bolívar. Caracas, Venezuela.
- Disponible en la página:
<http://www ldc.usb.ve/%7Emeza/ci-7521/LibroTeoConjYriarte.ps>

RECURSOS EN INTERNET

1. Material de apoyo del curso de “Matemáticas para las Ciencias de la Computación”, dictado por los profesores: Charles Leiserson, Eric Lehman, Srinivas Devadas y Albert Meyer. Curso liberado por el Massachusetts Institute of Technology (MIT) de los Estados Unidos.
 Disponible en:
<http://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-042j-mathematics-for-computer-science-spring-2005/>
2. Material de apoyo del libro: “Matemáticas Discretas y sus Aplicaciones” de Kenneth Rosen. Todo el material de apoyo puede ser libremente descargado desde la página oficial del libro en la editorial Mc Graw Hill:
http://highered.mcgraw-hill.com/sites/0072880082/student_view0/index.html
http://highered.mcgraw-hill.com/sites/0072880082/student_view0/chapter1/

http://highered.mcgraw-hill.com/sites/0072880082/student_view0/chapter2/
http://highered.mcgraw-hill.com/sites/0072880082/student_view0/chapter3/
http://highered.mcgraw-hill.com/sites/0072880082/student_view0/chapter4/
http://highered.mcgraw-hill.com/sites/0072880082/student_view0/chapter8/

3. Material de apoyo del libro: “Matemática Discreta” de Norman L. Biggs. El material de apoyo que se brinda puede ser libremente descargado desde la página oficial del libro en OXFORD University Press:

<http://www.oup.co.uk/academic/companion/mathematics/discretemath/>

4. Material de apoyo del curso de “Matemáticas Discretas I”, elaborado por el profesor Raúl Ernesto Gutiérrez de Piñerez. Curso liberado por la Universidad del Valle, Cali, Colombia.

Disponible en:

http://eisc.univalle.edu.co/materias/Matematicas_Discretas_1/

3.2. Programa de matemáticas computacionales II

El segundo curso de matemáticas computacionales contiene los temas que tradicionalmente se trabajan en un curso de Matemáticas Discretas II. El programa sugerido es el siguiente:

Propuesta de Programa para el SEGUNDO Curso de Matemáticas Computacionales En Programas de Estudio de Ingeniería de Sistemas

Semestre sugerido: Cuarto

Créditos sugeridos: 3

Intensidad horaria sugerida: 4 horas semanales

Requisitos sugeridos: Matemáticas Computacionales I y Programación II

Propuesta realizada por: Ing. Hugo Humberto Morales Peña

JUSTIFICACIÓN

Las Redes, la Investigación de Operaciones, el Análisis de Algoritmos y la Inteligencia Artificial, entre otros, se fundamentan en los temas trabajados en Matemáticas Discretas. Por este motivo, es de gran importancia que los estudiantes de Ingeniería de Sistemas reciban una bases solidas en estos temas.

OBJETIVOS

Objetivo General

Presentar algunas de las estructuras matemáticas discretas fundamentales para la computación, tales como, las Técnicas de Conteo y la Teoría de Grafos. Comprender su relación directa con áreas tales como Redes, Investigación de Operaciones y Análisis de Algoritmos.

Objetivos Específicos

- Calcular permutaciones y combinaciones de un conjunto e interpretar su significado en el contexto de una aplicación particular.
- Comprender la terminología básica de grafos y árboles.
- Modelar problemas usando grafos y árboles.
- Relacionar los conceptos de grafos y árboles con estructuras de datos y algoritmos.

METODOLOGÍA

Clases

El curso tendrá cuatro (4) horas de intensidad semanal separadas en dos bloques de estudio cada uno de dos horas.

Lecturas

Cada unidad del programa tendrá lecturas complementarias en el material de apoyo del curso, las lecturas le permitirán al estudiante obtener un panorama más amplio sobre el tema. La responsabilidad de estas lecturas corresponde directamente al estudiante.

Texto

Adicional al material de apoyo del curso, se tendrá como texto guía el libro: Discrete Mathematics and Its Applications, de Kenneth Rosen [R2007].

EVALUACIÓN

- | | |
|-----------------------------|------|
| ▪ Primer examen parcial | 25 % |
| ▪ Segundo examen parcial | 25 % |
| ▪ Examen final | 25 % |
| ▪ Tareas, talleres y quices | 25 % |

CONTENIDO DEL CURSO

Unidad 1 (3 semanas): Técnicas de Conteo

- Las reglas de la suma y el producto.
- Permutaciones.
- Permutaciones con repetición.
- Combinaciones.
- Combinaciones con repetición: Distribuciones.
- Coeficiente binomial.
- Teorema del binomio.
- El triángulo de Pascal.
- Identidades de los coeficientes binomiales.
- El principio de inclusión exclusión.
- El principio de inclusión y exclusión generalizado.
- El principio del palomar.

Unidad 2 (3 semanas): Relaciones de Recurrencia

- Introducción a las relaciones de recurrencia.
- Modelamiento de problemas con relaciones de recurrencia.
- Solución de problemas de conteo por medio de la utilización de relaciones de recurrencia.
- Solución de relaciones de recurrencia lineales homogéneas con coeficientes constantes.
- Solución de relaciones de recurrencia lineales no homogéneas con coeficientes constantes.

Unidad 3 (5 semanas): Introducción a la Teoría de Grafos

- Introducción a los grafos.
- Tipos de grafos.
- Modelaje de problemas por medio de grafos.
- Terminología de grafos.

- Grafos especiales (K_n , C_n , W_n).
- Grafos bipartitos.
- Representación de grafos:
 - Matrices de adyacencia.
 - Matrices de incidencia.
- Isomorfismo de grafos.
- Conectividad:
 - Camino entre un par de vertices en un grafo.
 - Conectividad en grafos no dirigidos.
 - Conectividad en grafos dirigidos.
 - Caminos e isomorfismos.
 - Conteo de caminos entre un par de vertices.
- Caminos y ciclos de Euler.
- Caminos y ciclos de Hamilton.
- El problema de la ruta más corta:
 - Algoritmo de Dijkstra para calcular la distancia mínima entre dos vertices de un grafo.
- El problema del vendedor viajero.
- Grafos planos.
- Coloración de grafos.

Unidad 4 (5 semanas): Árboles

- Introducción a los árboles.
- Árboles como modelos.
- Propiedades de los árboles.
- Aplicaciones de árboles:
 - Árboles de búsqueda binaria.
 - Árboles de decisión.
 - Árboles de juego.
- Recorridos en árboles:

- Recorrido en orden.
- Recorrido en pre-orden.
- Recorrido en post-orden.
- Algoritmo de ordenamiento que utiliza árboles de búsqueda binaria y el recorrido en orden.
- Árboles abarcadores:
 - Árbol abarcador generado por una búsqueda primero en profundidad.
 - Árbol abarcador generado por una búsqueda primero en amplitud.
 - Aplicaciones con retroceso (backtracking).
- Árboles abarcadores minimales:
 - Algoritmo de Prim para generar árboles abarcadores minimales.
 - Algoritmo de Kruskal para generar árboles abarcadores minimales.

PARCELACIÓN DEL CURSO POR CLASES

1. Presentación del curso, forma y fechas de evaluación. Las reglas de la suma y el producto.
2. Permutaciones. Permutaciones con repetición.
3. Combinaciones. Combinaciones con repetición: Distribuciones.
4. Coeficiente binomial. Teorema del binomio. El triángulo de Pascal. Identidades de los coeficientes binomiales.
5. El principio de inclusión exclusión. El principio de inclusión y exclusión generalizado.
6. El principio del palomar. Entrega a los estudiantes del primer taller calificable por grupos.
7. Modelamiento de problemas de conteo por medio de relaciones de recurrencia.
8. Repaso del método de iteración para resolver relaciones de recurrencia lineales de primer orden. Solución de relaciones de recurrencia lineales homogéneas de segundo orden con coeficientes constantes.
9. Continuación solución de relaciones de recurrencia lineales homogéneas de segundo orden con coeficientes constantes.
10. Solución de relaciones de recurrencia lineales no homogéneas de segundo orden con coeficientes constantes. Entrega a los estudiantes del segundo taller calificable por grupos.

11. Continuación solución de relaciones de recurrencia lineales no homogéneas de segundo orden con coeficientes constantes.
12. Primera evaluación parcial.
13. Introducción a los grafos. Tipos de grafos. Modelaje de problemas por medio de grafos.
14. Terminología de grafos. Grafos especiales (K_n , C_n , W_n). Grafos bipartitos.
15. Representación de grafos por medio de matrices de adyacencia. Representación de grafos por medio de Matrices de incidencia.
16. Isomorfismo de grafos.
17. Camino entre un par de vertices en un grafo. Conectividad en grafos no dirigidos. Conectividad en grafos dirigidos.
18. Caminos e isomorfismos. Conteo de caminos entre un par de vertices. Entrega a los estudiantes del tercer taller calificable por grupos.
19. Caminos y ciclos de Euler.
20. Caminos y ciclos de Hamilton.
21. Algoritmo de Dijkstra para calcular la distancia mínima entre dos vertices de un grafo. Entrega a los estudiantes del cuarto taller calificable por grupos.
22. El problema del vendedor viajero. Grafos planos. Coloración de grafos.
23. Segunda evaluación parcial.
24. Introducción a los árboles. Árboles como modelos. Propiedades de los árboles.
25. Aplicación de árboles: Árboles de búsqueda binaria.
26. Aplicaciones de árboles: Árboles de decisión y Árboles de juego.
27. Recorridos en árboles: en orden, en pre-orden y en post-orden.
28. Algoritmo de ordenamiento que utiliza árboles de búsqueda binaria y el recorrido en orden. Entrega a los estudiantes del quinto taller calificable por grupos.
29. Árbol abarcador generado por una búsqueda primero en profundidad.
30. Árbol abarcador generado por una búsqueda primero en amplitud.
31. Aplicaciones con retroceso (backtracking). Entrega a los estudiantes del sexto taller calificable por grupos.
32. Algoritmo de Prim para generar árboles abarcadores minimales. Algoritmo de Kruskal para generar árboles abarcadores minimales.
33. Evaluación final.

BIBLIOGRAFÍA

- [G2003] Grimaldi, Ralph. 2003. “Discrete and Combinatorial Mathematics”, Five Edition. Pearson.
- [J2008] Johnsonbaugh, Richard. 2008. “Discrete Mathematics”. Seven Edition. Prentice Hall.
- [J2009] Jiménez Murillo, José A.. 2009. “Matemáticas para la Computación”. Alfaomega.
- [KBR2008] Kolman, Bernard, Busby, Robert y Ross, Sharon C.. 2008. “Estructuras de Matemáticas Discretas para la Computación”, Sexta edición. Prentice Hall.
- [R2007] Rosen, Kenneth H.. 2007. “Discrete Mathematics and Its Applications”, Six edition. Mc Graw Hill.
- [RW2002] Ross, Kenneth A. y Wright, Charles R.. 2002. “Matemáticas Discretas”, Quinta edición. Prentice Hall.
- [Y1996] Yriarte, Vicente. 1996. “Elementos de Teoría Combinatoria”. Universidad Simón Bolívar. Caracas, Venezuela.
- Disponible en la página del autor:
<http://www ldc.usb.ve/~yriarte/d3index.html>

RECURSOS EN INTERNET

1. Material de apoyo del libro: “Matemáticas Discretas y sus Aplicaciones” de Kenneth Rosen. Todo el material de apoyo puede ser libremente descargado desde la página oficial del libro en la editorial Mc Graw Hill:
http://highered.mcgraw-hill.com/sites/0072880082/student_view0/index.html
http://highered.mcgraw-hill.com/sites/0072880082/student_view0/chapter5/
http://highered.mcgraw-hill.com/sites/0072880082/student_view0/chapter7/
http://highered.mcgraw-hill.com/sites/0072880082/student_view0/chapter9/
http://highered.mcgraw-hill.com/sites/0072880082/student_view0/chapter10/
2. Material de apoyo del curso de “Matemáticas Discretas II”, dictado por el profesor Orlando Arboleda Molina. Curso liberado por la Universidad del Valle, Cali, Colombia.
 Disponible en:
http://eisc.univalle.edu.co/materias/Matematicas_Discretas_2/
3. Las notas de clase del curso de “Teoría de grafos”, de los profesores Raúl Gómez Marín y Andrés Sicard Ramírez, de la Universidad EAFIT de la ciudad de Medellín en Colombia.
 Disponible en:
<http://www1.eafit.edu.co/asicard/pubs/grafos.pdf>

3.3. Programa de matemáticas computacionales III

El tercer curso de matemáticas computacionales contiene los temas que tradicionalmente se trabajan en un curso de Teoría de Autómatas y Lenguajes Formales. El programa sugerido es el siguiente:

Propuesta de Programa para el TERCER Curso de Matemáticas Computacionales En Programas de Estudio de Ingeniería de Sistemas

Semestre sugerido: Quinto

Créditos sugeridos: 4

Intensidad horaria sugerida: 6 horas semanales

Requisitos sugeridos: Matemáticas Computacionales II

Propuesta realizada por: Ing. Hugo Humberto Morales Peña

JUSTIFICACIÓN

Los Compiladores, la Ingeniería del Software y la Inteligencia Artificial, entre otros, se fundamentan en sus aspectos operativos en conceptos que devienen de la teoría informática, y que son debidamente conjuntados por la teoría de Lenguajes Formales, Máquinas de Estados y Gramáticas Formales. Por consiguiente, es de vital importancia que los estudiantes accedan a este conocimiento desde una perspectiva puramente formal.

OBJETIVOS

Objetivo General

Explorar las contribuciones de las Matemáticas, la Ingeniería y la Lingüística al desarrollo de las Ciencias de la Computación así como sus implicaciones en el diseño de compiladores y de computadores, a través de la asimilación de los conceptos fundamentales de Lenguajes Formales, Máquinas de Estados y Gramáticas Formales.

Objetivos Específicos

- Definir formalmente el concepto de lenguaje y verificar que este concepto contiene el de lenguaje de programación.
- Caracterizar algunos modelos de máquinas de estados como lo son los autómatas finitos, los autómatas de pila y las máquinas de Turing en función de su capacidad para reconocer lenguajes.

- Definir formalmente el concepto de gramática y clasificar el conjunto de todas las gramáticas en función de su capacidad para generar lenguajes (Jerarquía de Chomsky).
- Establecer la relación directa que hay entre expresiones regulares, lenguajes regulares, gramáticas regulares y autómatas finitos.
- Establecer la relación directa que hay entre lenguajes independientes del contexto, gramáticas independientes del contexto y autómatas de pila.
- Establecer la relación directa que hay entre lenguajes recursivos, gramáticas de estructura de frase y máquinas de Turing.

METODOLOGÍA

Clases y trabajo en grupo

El curso tendrá seis (6) horas de intensidad semanal separadas en tres bloques de estudio cada uno de dos horas.

Las clases serán teóricas (6 horas/semana), donde a los estudiantes periódicamente se les dejará talleres para que sean trabajados en grupos de tres estudiantes.

El trabajo en clase se centrará en presentar los temas en forma magistral, resolver las dudas encontradas por los estudiantes durante la preparación del material, la solución de ejercicios que se hayan asignado, pero sobre todo en discutir nuevos ejercicios que permitan alcanzar mayor claridad en cada tema.

Lecturas

El estudiante debe leer con antelación a cada clase el material de la misma, así como haber realizado la mayor cantidad posible de los ejercicios propuestos en ese material bibliográfico. Como apoyo el profesor podrá publicar material de apoyo en una página Web y/o entregarlo en conferencias.

Texto

El texto guía de este curso es el libro de: Teoría de Autómatas y Lenguajes Formales, de Dean Kelly [K1995].

EVALUACIÓN

- | | |
|-----------------------------|------|
| ▪ Primer examen parcial | 25 % |
| ▪ Segundo examen parcial | 25 % |
| ▪ Examen final | 25 % |
| ▪ Tareas, talleres y quices | 25 % |

CONTENIDO DEL CURSO

Unidad 0 (Una semana):

Herramientas Matemáticas (repaso)

- Conjuntos.
- Funciones.
- Cardinalidad de conjuntos infinitos.
- Relaciones.
- Relaciones de equivalencia.
- Clases de equivalencia y particiones

Unidad 1 (Una semana):

Alfabetos, Palabras y Lenguajes

- Alfabetos. Operaciones entre alfabetos.
- Palabras (o cadenas). Operaciones entre cadenas.
- Lenguajes. Operaciones entre lenguajes.
- Lenguaje universal (Σ^*).
- Cardinalidad del lenguaje universal ($|\Sigma^*|$).

Unidad 2 (Una semana):

Lenguajes Regulares y Expresiones Regulares

- Lenguajes regulares.
- Expresiones regulares.
- Cerradura de Kleene y cerradura positiva.
- Equivalencia entre expresiones regulares.
- Simplificación de expresiones regulares.

Unidad 3 (3 semanas):

Autómatas Finitos

- Autómatas Finitos Deterministas (AFD).
- Autómatas Finitos No Deterministas (AFND).
- λ -Transiciones en AFND.

- Algoritmo de eliminación de λ -Transiciones en AFND.
- Equivalencia entre AFND y AFD.
- Algoritmo para transformar un AFND en un AFD.
- Generación de autómatas finitos a partir de expresiones regulares.
- Lema de Arden
- Expresión regular que representa el conjunto de palabras que pertenecen al lenguaje regular reconocido por un autómata finito.
- Algoritmo de minimización de estados de un AFD
- Modelos de Mealy y Moore

Unidad 4 (5 semanas):
Gramáticas Formales

- Gramáticas regulares
 - Equivalencia entre expresiones regulares, lenguajes regulares y gramáticas regulares.
 - Algoritmo para obtener una gramática regular a partir de un autómata finito.
 - Algoritmo para obtener un AFND a partir de una gramática regular.
 - Algoritmo para la eliminación de la recursividad por la izquierda en una gramática regular.
 - Lema del Bombeo para determinar si un lenguaje no es regular.
- Gramáticas independientes del contexto
 - Árboles de derivación o de análisis de ambigüedad.
 - El problema de la ambigüedad en la gramáticas independientes del contexto.
 - Simplificación de gramáticas independientes del contexto.
 - Algoritmo de eliminación de no terminales que no generan cadenas de terminales.
 - Algoritmo de eliminación de producciones inalcanzables desde el símbolo inicial de la gramática.
 - Algoritmo de eliminación de λ -producciones.
 - Algoritmo de eliminación de producciones unitarias.
 - Forma normal de Chomsky.
 - Forma normal de Greibach.
- Gramáticas de estructura de frase.
- Gramáticas dependientes del contexto.

**Unidad 5 (2.5 semanas):
Autómatas de Pila**

- Definición formal de autómata de pila determinístico.
- Representación gráfica de autómatas de pila determinísticos.
- Lenguaje reconocido por un autómata de pila determinístico.
- Construcción de autómatas de pila determinísticos a partir de un lenguaje independiente del contexto.
- Definición formal de autómata de pila no determinístico.
- Lenguaje reconocido por un autómata de pila no determinístico.
- Construcción de autómatas de pila no determinísticos a partir de un lenguaje independiente del contexto.
- Algoritmo para generar un autómata de pila no determinístico a partir de una gramática independiente del contexto.
- Equivalencia entre lenguajes independientes del contexto, gramáticas independientes del contexto y autómatas de pila.

**Unidad 6 (2.5 semanas):
Máquinas de Turing**

- Definición formal de una Máquina de Turing.
- Representación gráfica de una Máquina de Turing.
- Notaciones de descripciones instantáneas para hacerle seguimiento al reconocimiento de una palabra en una Máquina de Turing.
- Lenguaje reconocido por una Máquina de Turing.
- Construcción de una Máquina de Turing a partir de un lenguaje de estructura de frase.
- Definición formal de una Máquina de Turing multicinta.
- Representación gráfica de una Máquina de Turing multicinta.
- Notaciones de descripciones instantáneas para hacerle seguimiento al reconocimiento de una palabra en una Máquina de Turing multicinta.
- Lenguaje reconocido por una Máquina de Turing multicinta.
- Construcción de una Máquina de Turing multicinta a partir de un lenguaje de estructura de frase.

PARCELACIÓN DEL CURSO POR CLASES

Semanalmente se dictan tres clases, cada una de dos horas.

1. Presentación del curso, forma y fechas de evaluación. Repaso de teoría de conjuntos.
2. Repaso de funciones y relaciones.
3. Repaso de relaciones de equivalencia, clase de equivalencia y particiones.
4. Alfabetos y operaciones entre alfabetos. Palabras (o cadenas) y operaciones entre cadenas.
5. Lenguajes y operaciones entre lenguajes.
6. Lenguaje universal (Σ^*). Cardinalidad del lenguaje universal ($|\Sigma^*|$).
7. Lenguajes regulares.
8. Expresiones regulares. Cerradura de Kleene y cerradura positiva.
9. Equivalencia entre expresiones regulares. Simplificación de expresiones regulares.
10. Autómatas finitos deterministas.
11. Autómatas finitos no deterministas.
12. λ -transiciones en AFND. Algoritmo de eliminación de λ -transiciones en AFND.
13. Equivalencia entre AFND y AFD. Algoritmo para transformar un AFND en un AFD.
14. Generación de autómatas finitos a partir de expresiones regulares.
15. Lema de Arden. Expresión regular que representa el conjunto de palabras que pertenecen al lenguaje regular reconocido por un autómata finito.
16. Algoritmo de minimización de estados de un AFD.
17. Modelos de Mealy y Moore.
18. Primera evaluación parcial.
19. Equivalencia entre expresiones regulares, lenguajes regulares y gramáticas regulares.
20. Algoritmo para obtener una gramática regular a partir de un autómata finito.
21. Algoritmo para obtener un AFND a partir de una gramática regular.
22. Algoritmo para la eliminación de la recursividad por la izquierda en una gramática regular.

23. Lema del Bombeo para determinar si un lenguaje no es regular.
24. Árboles de derivación o de análisis de ambigüedad. El problema de la ambigüedad en la gramáticas independientes del contexto.
25. Simplificación de gramáticas independientes del contexto. Algoritmo de eliminación de no terminales que no generan cadenas de terminales.
26. Algoritmo de eliminación de producciones inalcanzables desde el símbolo inicial de la gramática.
27. Algoritmo de eliminación de λ -producciones.
28. Algoritmo de eliminación de producciones unitarias.
29. Forma normal de Chomsky.
30. Forma normal de Greibach.
31. Gramáticas de estructura de frase.
32. Gramáticas dependientes del contexto.
33. Segunda evaluación parcial.
34. Definición formal de autómata de pila determinístico. Representación gráfica de autómatas de pila determinísticos.
35. Lenguaje reconocido por un autómata de pila determinístico.
36. Construcción de autómatas de pila determinísticos a partir de un lenguaje independiente del contexto.
37. Definición formal de autómata de pila no determinístico. Lenguaje reconocido por un autómata de pila no determinístico.
38. Construcción de autómatas de pila no determinísticos a partir de un lenguaje independiente del contexto.
39. Algoritmo para generar un autómata de pila no determinístico a partir de una gramática independiente del contexto.
40. Equivalencia entre lenguajes independientes del contexto, gramáticas independientes del contexto y autómatas de pila.
41. Definición formal de una Máquina de Turing. Representación gráfica de una Máquina de Turing.
42. Notaciones de descripciones instantáneas para hacerle seguimiento al reconocimiento de una palabra en una Máquina de Turing.

43. Lenguaje reconocido por una Máquina de Turing.
44. Construcción de una Máquina de Turing a partir de un lenguaje de estructura de frase.
45. Definición formal de una Máquina de Turing multicinta. Representación gráfica de una Máquina de Turing multicinta.
46. Notaciones de descripciones instantáneas para hacerle seguimiento al reconocimiento de una palabra en una Máquina de Turing multicinta. Lenguaje reconocido por una Máquina de Turing multicinta.
47. Construcción de una Máquina de Turing multicinta a partir de un lenguaje de estructura de frase.
48. Autómatas finitos versus autómatas de pila versus Máquinas de Turing.
49. Evaluación final.

BIBLIOGRAFÍA

- [dC2004] de Castro Korgi, Rodrigo. 2004. “Teoría de la Computación: lenguajes, autómatas, gramáticas”, Universidad Nacional de Colombia, Facultad de Ciencias, Departamento de Matemáticas.
- [GS2002] Gómez Marín, Raúl y Sicard Ramírez, Andrés. 2002. “Informática Teórica. Elementos Propedéuticos”. Fondo Editorial de la Universidad EAFIT, Medellín, Colombia.
- Disponible en la página del autor:
<http://www1.eafit.edu.co/asicard/pubs/informatica-teorica.pdf>
- [HMU2002] Hopcroft, J. E., Motwani, R. y Ullman, J. D.. 2002. “Introducción a la Teoría de Autómatas, Lenguajes y Computación”, Segunda Edición. Pearson Educación S.A.
- [K1995] Kelly, Dean. 1995. “Teoría de Autómatas y Lenguajes Formales”. Prentice Hall.
- [S1996] Sipser, M.. 1996. “Introduction to the Theory of Computation”. Thomson/Brooks Cole.
- [VMB1997] Viñuela, P. I., Martínez F., P. y Borrajo M., D.. 1997. “Lenguajes Gramáticas y Autómatas un Enfoque Práctico”. Addison-Wesley Iberoamericana España, S.A.

RECURSOS EN INTERNET

1. Material de apoyo del curso de “Autómatas, Computabilidad y Complejidad”, dictado por la profesora Nancy Lynch. Curso liberado por el Massachusetts Institute of Technology (MIT) de los Estados Unidos.

Disponible en:

<http://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-045j-automata-computability-and-complexity-spring-2005/>

2. Material de apoyo del curso de “Teoría de la Computación”, dictado por el profesor Michael Sipser. El libro guía de este curso es “Introduction to the Theory of Computation” [S1996] quien tiene como autor al mismo profesor de la materia. Curso liberado por el Massachusetts Institute of Technology (MIT) de los Estados Unidos.

Disponible en:

<http://ocw.mit.edu/courses/mathematics/18-404j-theory-of-computation-fall-2006/>

3. Material de apoyo del curso de “Computabilidad y Lenguajes Formales”, dictado por la profesora Gloria Inés Álvarez. Curso liberado por la Pontificia Universidad Javeriana de Cali en Colombia.

Disponible en:

<http://cic.puj.edu.co/wiki/doku.php?id=materias:computabilidad>

4. Software JFLAP. Software que permite trabajar gráficamente con autómatas finitos, autómatas de pila y Máquinas de Turing.

Disponible en:

<http://www.cs.duke.edu/csed/jflap/>

3.4. Programa de matemáticas computacionales IV

El cuarto curso de matemáticas computacionales contiene los temas que tradicionalmente se trabajan en un curso de Análisis y Diseño de Algoritmos. El programa sugerido es el siguiente:

Propuesta de Programa para el CUARTO Curso de Matemáticas Computacionales En Programas de Estudio de Ingeniería de Sistemas

Semestre sugerido: Sexto

Créditos sugeridos: 4

Intensidad horaria sugerida: 4 horas semanales

Requisitos sugeridos: Matemáticas Computacionales III

Propuesta realizada por: Ing. Hugo Humberto Morales Peña

JUSTIFICACIÓN

El curso de Análisis y Diseño de Algoritmos es el eslabón que finaliza y une dos líneas de trabajo bien diferenciadas en la Ingeniería de Sistemas, la primera, la línea que la componen todos los cursos de programación, y, la segunda, la línea que la componen todos los cursos de matemáticas. De esta forma, el análisis de algoritmos se apoya en fundamentos matemáticos para determinar la complejidad computacional de los algoritmos, y así poder determinar que tan buenos o malos son. El análisis de algoritmos es una herramienta que todo ingeniero de sistemas debe saber manejar; herramienta que le sirve para poder determinar entre diferentes algoritmos para resolver una misma tarea, cual de ellos es el mejor con respecto a los aspectos que se quieren optimizar.

OBJETIVOS

Objetivos Generales

- Manejar técnicas fundamentales de análisis y diseño de algoritmos para comprender la naturaleza de los problemas tan independientemente como sea posible de aspectos de implementación (tanto hardware como software) y donde la solución sea eficiente.
- Conocer las clases de problemas computacionales que no tienen soluciones algorítmicas eficientes en la práctica.

Objetivos Específicos

- Comprender el compromiso de complejidad entre espacio y tiempo usado por un algoritmo.
- Conocer la clasificación estándar de problemas en clases de complejidad.
- Manejar diferentes técnicas de diseño de algoritmos tales como técnicas voraces, ingenuas, de dividir y conquistar, programación dinámica.
- Utilizar la búsqueda secuencial, la búsqueda binaria y los algoritmos de ordenamiento para la solución de problemas computacionales.
- Escoger entre diferentes versiones de algoritmos el mejor bajo un conjunto de restricciones.
- Poder identificar problemas para los cuales no existen soluciones algorítmicas eficientes y caracterizarlos: problemas tratables e intratables.
- Comprender las clases P y NP, y el significado de la NP-completitud.
- Comprender demostraciones de la NP-completitud de algunos problemas.

METODOLOGÍA

Clases y uso del computador

El curso tendrá cuatro (4) horas de intensidad semanal separadas en dos bloques de estudio cada uno de dos horas.

Las clases serán teóricas (4 horas/semana), pero el curso tendrá un alto contenido práctico con el desarrollo de los miniproyectos que deben ser realizados en grupos de tres estudiantes.

Lecturas

El estudiante debe leer con antelación a cada clase el material de la misma (indicado entre corchetes al lado de cada tema del programa), así como haber realizado la mayor cantidad posible de los ejercicios propuestos en ese material bibliográfico.

Lenguaje de programación utilizado

Para el desarrollo de los miniproyectos se utilizará el lenguaje de programación Java.

Texto

El texto guía de este curso es el libro: “Introduction to Algorithms” de Thomas Cormen y otros [CLRS2001]. Los capítulos que se estudiarán de este libro son: 1, 2, 3, 4, 7, 8, 9, 16, 17 y 36. Un texto que sirve como base para algunos temas del curso es el libro: Discrete Mathematics and Its Applications, de Kenneth Rosen [R2007], dicho libro es de matemáticas discretas y trata de forma mas sencilla y pedagógica los temas de fundamentos matemáticos (series, sumatorias, aritmética modular y funciones piso y techo), relaciones de recurrencia, notación O , notación Theta (Θ), notación Omega (Ω), complejidad de un algoritmo y algunos de los algoritmos de ordenamiento por comparaciones.

EVALUACIÓN

■ Primer examen parcial	20 %
■ Segundo examen parcial	20 %
■ Examen final	20 %
■ Tareas, talleres y quices	8 %
■ Miniproyecto Uno	16 %
■ Miniproyecto Dos	16 %

CONTENIDO DEL CURSO

Unidad 1 (Una semana): Repaso de Herramientas Matemáticas

- Series (sucesiones)
- Sumatorias
- Inducción matemática
- Aritmética modular
- Propiedades de los logaritmos
- Propiedades de los exponentes y radicales
- Funciones piso y techo

Unidad 2 (4 semanas): Complejidad computacional

- Notación O .
- Notación Ω .
- Notación Θ .
- Complejidad de algoritmos iterativos.
- Utilización de las relaciones de recurrencia para calcular la complejidad de algoritmos recursivos.

Unidad 3 (3.5 semanas): Análisis de algoritmos de ordenamiento por comparaciones

- Insertion Sort
- Merge Sort
- Heap Sort
- Quick Sort

Unidad 4 (1.5 semanas): Análisis de algoritmos de ordenamiento en tiempo lineal

- Counting Sort
- Radix Sort
- Bucket Sort

Unidad 5 (Una semana): Búsqueda de un elemento en un vector

- Búsqueda lineal de un elemento en un vector.
- Búsqueda binaria de un elemento en un vector.

Unidad 6 (3 semanas): Estrategias algorítmicas de programación

- Técnica de programación ingenua.
- Técnica de programación de “divide y vencerás”.
- Técnica de programación dinámica.
- Técnica de programación voraz.

Unidad 7 (2 semanas): Problemas computacionalmente difíciles

- Las clases P y NP: Problemas tratables e intratables.
- NP-completitud.
- Problemas NP-completos.
- Reducción: Técnica para demostrar NP-completitud.

PARCELACIÓN DEL CURSO POR CLASES

Semanalmente se dictarán dos clases, cada una de dos horas.

1. Presentación del curso, forma y fechas de evaluación. Introducción al análisis y diseño de algoritmos.
2. Repaso de herramientas matemáticas (series, sumatorias, inducción matemática y aritmética modular).
3. Continuación del repaso de herramientas matemáticas (función piso, función techo, propiedades de los logaritmos, propiedades de los exponentes y radicales).
4. Notación O en complejidad computacional.
5. Notación O en complejidad computacional (continuación).
6. Notación Omega en complejidad computacional.
7. Notación Theta en complejidad computacional. Entrega a los estudiantes del primer taller para desarrollarlo en grupos.
8. Complejidad de algoritmos iterativos.

9. Utilización de las relaciones de recurrencia para calcular la complejidad de algoritmos recursivos.
10. Utilización de las relaciones de recurrencia para calcular la complejidad de algoritmos recursivos (continuación). Entrega a los estudiantes del segundo taller para desarrollarlo en grupos.
11. Primera evaluación parcial.
12. Análisis del algoritmo Insertion Sort de ordenamiento por comparaciones. Entrega a los estudiantes del enunciado del Miniproyecto 1.
13. Análisis del algoritmo Merge Sort de ordenamiento por comparaciones.
14. Análisis del algoritmo Merge Sort de ordenamiento por comparaciones (continuación).
15. Análisis del algoritmo Heap Sort de ordenamiento por comparaciones.
16. Análisis del algoritmo Heap Sort de ordenamiento por comparaciones (continuación).
17. Análisis del algoritmo Quick Sort de ordenamiento por comparaciones.
18. Análisis del algoritmo Quick Sort de ordenamiento por comparaciones (continuación).
19. Análisis del algoritmo Counting Sort de ordenamiento en tiempo lineal.
20. Análisis del algoritmo Radix Sort de ordenamiento en tiempo lineal.
21. Análisis del algoritmo Bucket Sort de ordenamiento en tiempo lineal.
22. Búsqueda lineal y búsqueda binaria de un elemento en un vector.
23. Segunda evaluación parcial.
24. Técnica de programación dinámica. Entrega a los estudiantes del enunciado del miniproyecto 2.
25. Técnica de programación dinámica (continuación).
26. Técnica de programación dinámica (continuación).
27. Programación recursiva con la técnica de memorización para resolver problemas de optimización.
28. Técnica de programación voraz.
29. Técnica de programación voraz (continuación).
30. Las clases P y NP: problemas tratables e intratables.

31. NP-completitud. Problemas NP-completos.
32. Reducción: Técnica para demostrar NP-completitud. Entrega a los estudiantes del tercer taller.
33. Evaluación final.

BIBLIOGRAFÍA

- [BB97] | Brassard, G., y Bratley P.. 1997. “Fundamentos de Algoritmia”. Prentice Hall.
- [C1993] | Cardoso, Rodrigo. 1993. “Verificación y Desarrollo de Programas”. Universidad de los Andes, Bogotá, Colombia.
- [CLRS2001] | Cormen, Thomas H., Leiserson, Charles E., Rivest, Ronald L., and Stein, Clifford. 2001. “Introduction to Algorithms”, Second Edition. The MIT Press, Cambridge, Massachusetts.
- [DPV2006] | Dasgupta, S., Papadimitriou, C. H., and Vazirani, U. V.. 2006. “Algorithms”. Mc Graw Hill Science/Engineering/Math.
- Disponible en la página del autor:
<http://www.cs.berkeley.edu/~vazirani/algorithms.html>
- [GV1998] | Guerequeta, Rosa y Vallecillo, Antonio. 1998. “Técnicas de Diseño de Algoritmos”. Servicio de Publicaciones de la Universidad de Málaga, España.
- Disponible en la página del autor:
<http://www.lcc.uma.es/~av/Libro/>
- [H97] | Hochbaum, D. S.. Editor. 1997. “Approximation Algorithms for NP-hard problems”. PWS Publishing Company.
- [R2007] | Rosen, Kenneth. H.. 2007. “Discrete Mathematics and Its Applications”, Six edition. Mc Graw Hill.
- [S1983] | Sedgewick, Robert. 1983. “Algorithms”. Addison Wesley.
- [S2002a] | Sedgewick, Robert. 2002. “Algorithms in Java: Parts 1-4: Fundamentals, Data Structures, Sorting and Searching”, Third Edition. Addison Wesley.
- [S2002b] | Sedgewick, Robert. 2002. “Algorithms in Java: Parts 5: Graph Algorithms”, Third Edition. Addison Wesley.
- [S98] | Skiena, S.. 1998. “The Algorithms Design Manual”. Springer-Verlag.

RECURSOS EN INTERNET

1. Material multimedia del curso de “Introducción a los Algoritmos” que realmente es un curso de Análisis de Algoritmos, dictado por el profesor Charles E. Leiserson, donde se utiliza como texto guía el libro: “Introduction to Algorithms” del cual es coautor [CLRS2001]. Curso liberado por el Massachusetts Institute of Technology (MIT) de los Estados Unidos.

Disponible en:

<http://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-046j-introduction-to-algorithms-sma-5503-fall-2005/>

2. Material de apoyo del curso de “Teoría de la Complejidad Avanzada”, dictado por el profesor Daniel Spielman. Curso liberado por el Massachusetts Institute of Technology (MIT) de los Estados Unidos.

Disponible en:

<http://ocw.mit.edu/courses/mathematics/18-405j-advanced-complexity-theory-fall-2001/>

3. Material de apoyo del curso de “Algoritmos y Estructuras de Datos”, dictado por los profesores Robert Sedgewick y Kevin Wayne. Los textos guías de este curso son “Algorithms”, “Algorithms in Java” del mismo profesor de la materia, Robert Sedgewick ([S1983], [S2002a] y [S2002b]). Curso liberado por la Universidad de Princeton de los Estados Unidos.

Disponible en:

<http://www.cs.princeton.edu/courses/archive/spring08/cos226/info.html>

4. Material de apoyo del curso de “Fundamentos de Análisis y Diseño de Algoritmos - FADA”, dictado por el profesor Juan Francisco Díaz. Curso liberado por la Universidad del Valle sede Cali en Colombia.

Disponible en:

<http://eisc.univalle.edu.co/materias/FADA/>

5. Material de apoyo del curso de “Análisis y Diseño de Algoritmos”, dictado por el profesor Andrés Becerra Sandoval. Curso liberado por la Pontificia Universidad Javeriana de Cali en Colombia.

Disponible en:

http://cic.puj.edu.co/wiki/doku.php?id=materias:analisis_y_diseno_de_algoritmos

Capítulo 4

Estrategia utilizada para la construcción del material de apoyo

La estrategia utilizada para la construcción del material de apoyo consiste en presentar los temas de forma tal que estos sean la base o fundamentación necesaria para el fácil entendimiento de temas posteriores por parte de los estudiantes. Se ha utilizado una estrategia pensada principalmente en el aprendizaje por parte de los estudiantes, más que sobre la enseñanza por parte del profesor. Pero lo primero conlleva a lo segundo, entre más fácil aprendan los estudiantes ... más fácil se vuelve la enseñanza por parte del profesor.

A continuación se presentará la forma de como la suma de términos de una serie geométrica puede ser utilizada para calcular la fórmula que representa la siguiente suma de términos:

$$\frac{1}{2^1} + \frac{2}{2^2} + \frac{3}{2^3} + \frac{4}{2^4} + \frac{5}{2^5} + \dots + \frac{n}{2^n}.$$

4.1. Primero, el trabajo con sucesiones (o series)

Definición: Una sucesión es una función del conjunto de los números naturales o del conjunto de los números enteros positivos a un conjunto S . Se usa la notación S_n para denotar la imagen del número n en el conjunto S , de esta misma forma, S_n representa al término ubicado en la posición n de la sucesión. Se usa la notación $\{S_n\}$ para denotar todo el rango de la función. Las sucesiones son descritas listando los términos de la sucesión en el orden como se va incrementando el subíndice.

Ejemplo 1:

Considerar la sucesión $\{S_n\}$, donde $S_n = \frac{n}{2^n}$.

Los elementos de la sucesión, comenzando en S_1 , son:

$$S_1 = \frac{1}{2^1}$$

$$S_2 = \frac{2}{2^2}$$

$$S_3 = \frac{3}{2^3}$$

$$S_4 = \frac{4}{2^4}$$

$$S_5 = \frac{5}{2^5}$$

$$\vdots$$

$$S_n = \frac{n}{2^n}$$

de esta forma se tiene que la lista de términos de la sucesión es:

$$\frac{1}{2^1}, \frac{2}{2^2}, \frac{3}{2^3}, \frac{4}{2^4}, \frac{5}{2^5}, \dots, \frac{n}{2^n}$$

Nota: El ejemplo anterior no es fortuito, de antemano se quiere por parte del estudiante que conozca bien la sucesión de términos que posteriormente se van a sumar.

4.2. Segundo, el trabajo con la serie (o progresión) geométrica

Definición: Una progresión geométrica es una sucesión de la forma:

$$a, a \cdot r, a \cdot r^2, a \cdot r^3, \dots, a \cdot r^n$$

donde el término inicial a y la razón constante r son números reales.

Ejemplo 2:

La sucesión $\{S_n\}$ con $S_n = \left(\frac{1}{2}\right)^n$ es una progresión geométrica con término inicial $a = 1$ y razón constante $r = \frac{1}{2}$.

Los elementos de la sucesión, comenzando en $n = 0$, son:

$$S_0 = \left(\frac{1}{2}\right)^0 = 1$$

$$S_1 = \left(\frac{1}{2}\right)^1 = \frac{1}{2}$$

$$S_2 = \left(\frac{1}{2}\right)^2 = \frac{1}{4}$$

$$S_3 = \left(\frac{1}{2}\right)^3 = \frac{1}{8}$$

$$S_4 = \left(\frac{1}{2}\right)^4 = \frac{1}{16}$$

$$\vdots$$

$$S_n = \left(\frac{1}{2}\right)^n = \frac{1}{2^n}$$

de esta forma se tiene que la lista de términos de la sucesión es:

$$1, \frac{1}{2}, \frac{1}{4}, \frac{1}{8}, \frac{1}{16}, \dots, \frac{1}{2^n}$$

4.3. Tercero, el trabajo con sumatorias

La notación de sumatoria es usada para representar la suma de los términos $S_m, S_{m+1}, S_{m+2}, \dots, S_n$ de la sucesión $\{S_n\}$.

Se usa la notación $\sum_{i=m}^n S_i$ para representar $S_m + S_{m+1} + S_{m+2} + \dots + S_n$.

Ejemplo 3:

La suma de los primeros 50 términos de la sucesión $\{S_n\}$ donde el n -ésimo término de la sucesión está definido por $S_n = \frac{n}{2^n}$ puede ser representada por medio de sumatorias como:

$$\sum_{i=1}^{50} \frac{i}{2^i} = \frac{1}{2^1} + \frac{2}{2^2} + \frac{3}{2^3} + \frac{4}{2^4} + \dots + \frac{50}{2^{50}}$$

Teorema:

Si a y r son números reales con $r \neq 0$, entonces:

$$\sum_{i=0}^n a \cdot r^i = \begin{cases} a \cdot (n+1) & \text{si } r = 1 \\ \frac{a \cdot r^{n+1} - a}{r - 1} & \text{si } r \neq 1 \end{cases}$$

Demostración:

Se debe demostrar cada uno de los dos casos del teorema de forma independiente, si los dos casos se cumplen entonces queda demostrada la validez del teorema

■ Caso donde $r = 1$

$$\sum_{i=0}^n a \cdot 1^i = \sum_{i=0}^n a \cdot 1 = \sum_{i=0}^n a = \underbrace{a + a + a + a + \dots + a}_{n+1 \text{ veces}} = a \cdot (n+1).$$

Queda demostrado el caso.

■ **Caso donde $r \neq 1$**

Sea $S = a + a \cdot r + a \cdot r^2 + a \cdot r^3 + \dots + a \cdot r^n$, al multiplicar a ambos lados de la igualdad por $-r$ se sigue conservando la igualdad, donde se obtiene: $-S \cdot r = -a \cdot r - a \cdot r^2 - a \cdot r^3 - a \cdot r^4 - \dots - a \cdot r^{n+1}$, al sumar ambas igualdades se tiene:

$$\begin{array}{r}
 S = a + a \cdot r + a \cdot r^2 + a \cdot r^3 + \dots + a \cdot r^n \\
 -S \cdot r = -a \cdot r - a \cdot r^2 - a \cdot r^3 - a \cdot r^4 - \dots - a \cdot r^n - a \cdot r^{n+1} \\
 \hline
 S - S \cdot r = a - a \cdot r^{n+1}
 \end{array}$$

$$S \cdot (1 - r) = a - a \cdot r^{n+1}$$

$$S = \frac{a - a \cdot r^{n+1}}{1 - r}$$

$$S = \frac{-1}{-1} \cdot \frac{a - a \cdot r^{n+1}}{1 - r}$$

$$S = \frac{-a + a \cdot r^{n+1}}{-1 + r}$$

$$S = \frac{a \cdot r^{n+1} - a}{r - 1}$$

Como S es igual a $a + a \cdot r + a \cdot r^2 + a \cdot r^3 + \dots + a \cdot r^n$ entonces $a + a \cdot r + a \cdot r^2 + a \cdot r^3 + \dots + a \cdot r^n = \frac{a \cdot r^{n+1} - a}{r - 1}$, para $n \in \mathbb{Z}^+$.

Queda demostrado el caso.

Como se cumplen todos los casos del teorema entonces queda demostrada la validez del teorema de la suma de términos de la serie o progresión geométrica.

Ejemplo 4:

Obtener el resultado de la siguiente suma de términos:

$$\sum_{i=0}^{1000} \left(\frac{1}{2}\right)^i = \left(\frac{1}{2}\right)^0 + \left(\frac{1}{2}\right)^1 + \left(\frac{1}{2}\right)^2 + \left(\frac{1}{2}\right)^3 + \dots + \left(\frac{1}{2}\right)^{1000}.$$

Los términos que se están sumando pertenecen a una serie geométrica, donde el primer

término es $a = 1$, la razón constante es $r = \frac{1}{2}$ y la potencia más grande es $n = 1000$, de esta forma se tiene que:

$$\sum_{i=0}^{1000} \left(\frac{1}{2}\right)^i = \frac{1 \cdot \left(\frac{1}{2}\right)^{1000+1} - 1}{\frac{1}{2} - 1} = \frac{\left(\frac{1}{2}\right)^{1001} - 1}{-\frac{1}{2}} = \frac{\frac{1}{2^{1001}} - 1}{-\frac{1}{2}} = 2 - \frac{1}{2^{1000}}$$

4.4. Cuarto, el uso de soluciones de sumatorias conocidas para calcular soluciones de sumatorias no conocidas

Ejemplo 5:

Obtener una fórmula que sea la solución de la siguiente suma de términos:

$$\sum_{i=1}^n \frac{i}{2^i} = \frac{1}{2^1} + \frac{2}{2^2} + \frac{3}{2^3} + \cdots + \frac{n}{2^n}, \text{ para } n \in \mathbb{Z}^+.$$

Los términos que se están sumando no pertenecen originalmente a una serie geométrica, pero se puede hacer un manejo de los términos para que la solución de la sumatoria de términos de la serie geométrica pueda ser utilizada. El manejo es el siguiente:

$$\begin{aligned} \sum_{i=1}^n \frac{i}{2^i} &= \frac{1}{2^1} + \frac{2}{2^2} + \frac{3}{2^3} + \cdots + \frac{n}{2^n} \\ &= \frac{1}{2^1} + \left(\frac{1}{2^2} + \frac{1}{2^2}\right) + \left(\frac{1}{2^3} + \frac{1}{2^3} + \frac{1}{2^3}\right) + \cdots + \underbrace{\left(\frac{1}{2^n} + \frac{1}{2^n} + \frac{1}{2^n} + \cdots + \frac{1}{2^n}\right)}_{n \text{ veces}} \\ &= \left(\frac{1}{2^1} + \frac{1}{2^2} + \frac{1}{2^3} + \cdots + \frac{1}{2^n}\right) + \left(\frac{1}{2^2} + \frac{1}{2^3} + \frac{1}{2^4} + \cdots + \frac{1}{2^n}\right) + \\ &\quad \left(\frac{1}{2^3} + \frac{1}{2^4} + \frac{1}{2^5} + \cdots + \frac{1}{2^n}\right) + \cdots + \left(\frac{1}{2^n}\right) \\ &= \sum_{i=1}^n \left(\frac{1}{2}\right)^i + \sum_{i=2}^n \left(\frac{1}{2}\right)^i + \sum_{i=3}^n \left(\frac{1}{2}\right)^i + \cdots + \sum_{i=n}^n \left(\frac{1}{2}\right)^i \\ &= \left(\sum_{i=0}^n \left(\frac{1}{2}\right)^i - \sum_{i=0}^0 \left(\frac{1}{2}\right)^i\right) + \left(\sum_{i=0}^n \left(\frac{1}{2}\right)^i - \sum_{i=0}^1 \left(\frac{1}{2}\right)^i\right) + \end{aligned}$$

$$\begin{aligned}
& \left(\sum_{i=0}^n \left(\frac{1}{2} \right)^i - \sum_{i=0}^2 \left(\frac{1}{2} \right)^i \right) + \cdots + \left(\sum_{i=0}^n \left(\frac{1}{2} \right)^i - \sum_{i=0}^{n-1} \left(\frac{1}{2} \right)^i \right) \\
&= n \left(\sum_{i=0}^n \left(\frac{1}{2} \right)^i \right) - \left(\sum_{i=0}^0 \left(\frac{1}{2} \right)^i + \sum_{i=0}^1 \left(\frac{1}{2} \right)^i + \sum_{i=0}^2 \left(\frac{1}{2} \right)^i + \cdots + \sum_{i=0}^{n-1} \left(\frac{1}{2} \right)^i \right) \\
&= n \left(\frac{\left(\frac{1}{2} \right)^{n+1} - 1}{\frac{1}{2} - 1} \right) - \left(\frac{\left(\frac{1}{2} \right)^1 - 1}{\frac{1}{2} - 1} + \frac{\left(\frac{1}{2} \right)^2 - 1}{\frac{1}{2} - 1} + \frac{\left(\frac{1}{2} \right)^3 - 1}{\frac{1}{2} - 1} + \cdots + \frac{\left(\frac{1}{2} \right)^n - 1}{\frac{1}{2} - 1} \right) \\
&= n \left(\frac{\frac{1}{2^{n+1}} - 1}{-\frac{1}{2}} \right) - \left(\frac{\left(\frac{1}{2} \right)^1 - 1}{-\frac{1}{2}} + \frac{\left(\frac{1}{2} \right)^2 - 1}{-\frac{1}{2}} + \frac{\left(\frac{1}{2} \right)^3 - 1}{-\frac{1}{2}} + \cdots + \frac{\left(\frac{1}{2} \right)^n - 1}{-\frac{1}{2}} \right) \\
&= n \left(\frac{\frac{1}{2^{n+1}} - 1}{-\frac{1}{2}} \right) - \left(\frac{\left(\frac{1}{2} \right)^1 - 1 + \left(\frac{1}{2} \right)^2 - 1 + \left(\frac{1}{2} \right)^3 - 1 + \cdots + \left(\frac{1}{2} \right)^n - 1}{-\frac{1}{2}} \right) \\
&= n \left(\frac{\frac{-1+2^{n+1}}{2^{n+1}}}{\frac{1}{2}} \right) + 2 \left(\left(\left(\frac{1}{2} \right)^1 + \left(\frac{1}{2} \right)^2 + \left(\frac{1}{2} \right)^3 + \cdots + \left(\frac{1}{2} \right)^n \right) - n \right) \\
&= n \left(\frac{-1 + 2^{n+1}}{2^n} \right) + 2 \left(\left(\sum_{i=1}^n \left(\frac{1}{2} \right)^i \right) - n \right) \\
&= n \left(\frac{-1 + 2^{n+1}}{2^n} \right) - 2n + 2 \sum_{i=1}^n \left(\frac{1}{2} \right)^i \\
&= n \left(\frac{-1}{2^n} + \frac{2^{n+1}}{2^n} \right) - 2n + 2 \left(\sum_{i=0}^n \left(\frac{1}{2} \right)^i - \sum_{i=0}^0 \left(\frac{1}{2} \right)^i \right) \\
&= n \left(\frac{-1}{2^n} + 2 \right) - 2n + 2 \left(\frac{\left(\frac{1}{2} \right)^{n+1} - 1}{\frac{1}{2} - 1} - \frac{\left(\frac{1}{2} \right)^1 - 1}{\frac{1}{2} - 1} \right) \\
&= -\frac{n}{2^n} + 2n - 2n + 2 \left(\frac{\frac{1}{2^{n+1}} - \frac{2^{n+1}}{2^{n+1}}}{-\frac{1}{2}} - \frac{\frac{1}{2} - 1}{\frac{1}{2} - 1} \right)
\end{aligned}$$

$$\begin{aligned}
&= -\frac{n}{2^n} + 2 \left(\frac{\frac{-1+2^{n+1}}{2^{n+1}}}{\frac{1}{2}} - 1 \right) \\
&= -\frac{n}{2^n} + 2 \left(\frac{-1+2^{n+1}}{2^n} - 1 \right) \\
&= -\frac{n}{2^n} + 2 \left(-\frac{1}{2^n} + \frac{2^{n+1}}{2^n} - 1 \right) \\
&= -\frac{n}{2^n} + 2 \left(-\frac{1}{2^n} + 2 - 1 \right) \\
&= -\frac{n}{2^n} + 2 \left(-\frac{1}{2^n} + 1 \right) \\
&= -\frac{n}{2^n} - \frac{2}{2^n} + 2 \\
&= 2 - \frac{n}{2^n} - \frac{2}{2^n} \\
&= 2 - \frac{1}{2^n} (n+2) \\
&= 2 - \left(\frac{n+2}{2^n} \right)
\end{aligned}$$

Por lo tanto:

$$\sum_{i=1}^n \frac{i}{2^i} = \frac{1}{2^1} + \frac{2}{2^2} + \frac{3}{2^3} + \dots + \frac{n}{2^n} = 2 - \left(\frac{n+2}{2^n} \right)$$

4.5. Cinco, uso de inducción matemática como método validador

Ejemplo 6:

Demostrar o refutar utilizando la técnica de inducción matemática que:

$$\sum_{i=1}^n \frac{i}{2^i} = \frac{1}{2^1} + \frac{2}{2^2} + \frac{3}{2^3} + \dots + \frac{n}{2^n} = 2 - \frac{n+2}{2^n}, \text{ para } n \in \mathbb{Z}^+$$

Para el desarrollo de la demostración considerar que se tiene la proposición

$$p(n) = \sum_{i=1}^n \frac{i}{2^i} = \frac{1}{2^1} + \frac{2}{2^2} + \frac{3}{2^3} + \dots + \frac{n}{2^n} = 2 - \frac{n+2}{2^n}, \text{ para } n \in \mathbb{Z}^+$$

la proposición $p(n)$ tomará el valor verdadero (V_o) o falso (F_o) dependiendo de si se cumple o no la igualdad $\sum_{i=1}^n \frac{i}{2^i} = 2 - \frac{n+2}{2^n}$. Ahora se consideran los tres pasos de la técnica por inducción matemática:

Paso base $n = 1$:

$$p(1) = \underbrace{\sum_{i=1}^1 \frac{i}{2^i} = \frac{1}{2^1} = \frac{1}{2}}_{\text{resultado a partir de la sumatoria de términos}} = \underbrace{2 - \frac{1+2}{2^1} = 2 - \frac{3}{2} = \frac{4-3}{2} = \frac{1}{2}}_{\text{resultado a partir de la fórmula}}$$

como se obtiene el mismo resultado en la sumatoria de términos y en la fórmula que es la solución de la sumatoria entonces la proposición $p(1)$ es verdadera y la demostración continua en el paso inductivo.

Paso inductivo $n = k$:

$p(k) = \sum_{i=1}^k \frac{i}{2^i} = \frac{1}{2^1} + \frac{2}{2^2} + \frac{3}{2^3} + \dots + \frac{k}{2^k} = 2 - \frac{k+2}{2^k}$, se asume que la proposición $p(k)$ es verdadera, esto quiere decir, que se supone que se cumple la siguiente igualdad $\frac{1}{2^1} + \frac{2}{2^2} + \frac{3}{2^3} + \dots + \frac{k}{2^k} = 2 - \frac{k+2}{2^k}$.

Paso post-inductivo $n = k + 1$:

$$p(k+1) = \sum_{i=1}^{k+1} \frac{i}{2^i} = \underbrace{\frac{1}{2} + \frac{2}{2^2} + \frac{3}{2^3} + \dots + \frac{k}{2^k}}_{\text{Se reemplaza por su equivalente en el paso inductivo}} + \frac{k+1}{2^{k+1}} = 2 - \frac{k+1+2}{2^{k+1}}$$

*Se reemplaza por su equivalente
en el paso inductivo*

$$\begin{aligned} 2 - \frac{k+2}{2^k} + \frac{k+1}{2^{k+1}} &= 2 - \frac{k+3}{2^{k+1}} \\ 2 - \left(\frac{k+2}{2^k} \cdot \frac{2}{2} \right) + \frac{k+1}{2^{k+1}} &= 2 - \frac{k+3}{2^{k+1}} \\ 2 - \frac{2k+4}{2^{k+1}} + \frac{k+1}{2^{k+1}} &= 2 - \frac{k+3}{2^{k+1}} \\ 2 - \left[\frac{2k+4}{2^{k+1}} - \frac{k+1}{2^{k+1}} \right] &= 2 - \frac{k+3}{2^{k+1}} \\ 2 - \left[\frac{2k+4 - (k+1)}{2^{k+1}} \right] &= 2 - \frac{k+3}{2^{k+1}} \\ 2 - \left[\frac{2k+4 - k - 1}{2^{k+1}} \right] &= 2 - \frac{k+3}{2^{k+1}} \end{aligned}$$

$$2 - \left[\frac{k+3}{2^{k+1}} \right] = 2 - \frac{k+3}{2^{k+1}}$$
$$2 - \frac{k+3}{2^{k+1}} = 2 - \frac{k+3}{2^{k+1}}$$

Se cumple la igualdad, por lo tanto la proposición $p(k+1)$ es verdadera, como se cumplen los tres pasos de técnica de demostración por inducción matemática entonces queda demostrada la validez de la solución de la sumatoria originalmente planteada.

En un documento anexo por aparte de 284 hojas se encuentra por completo el “Material de apoyo del primer curso de matemáticas computacionales”, el cual respeta las directrices trabajadas en este capítulo.

Capítulo 5

Estrategia utilizada para la construcción del banco de preguntas tipo ECAES

Dependiendo del lector éste es el capítulo más controversial e innecesario (y podría asegurarse, que debería de quitarse), o por el contrario, el lector considere que éste es el aporte más significativo de toda la investigación, que podría dar pie a una línea de trabajo en la construcción de preguntas tipo ECAES en temas de diferente indole.

Éste capítulo ha demandado muchas horas de investigación por parte del autor, no ha sido sencillo, para construir bancos de preguntas realmente interesantes (y útiles) para Ingeniería de Sistemas, el autor ha presentado (como egresado) los ECAES de Ingeniería de Sistemas durante los años 2004, 2005, 2006, 2007, 2008 y 2009, obteniendo los puntajes 126.9, 136.63, 135.66, 141.6, 142.08 y 123.8 respectivamente; puntajes que en cuatro de las seis presentaciones le permitirían a un estudiante activo estar en los diez mejores ECAES de Ingeniería de Sistemas a nivel nacional.

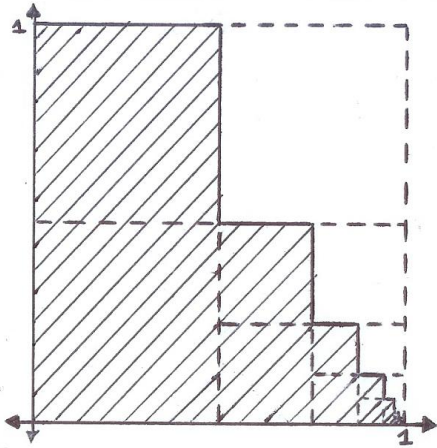
Hay muchas formas de enseñar un tema, una de ellas, posiblemente la más utilizada por parte de los docentes, es documentarse muy bien del tema, apropiarse de él lo mejor posible y utilizar alguna de las tantas metodologías existentes para enseñar el tema; y otra que consiste en adicionarle a todo lo anterior la experiencia del docente vivida en el tema, obviamente esta es una visión más de ingeniero que de matemático. En una preparación de estudiantes para un examen de calidad, prácticamente la totalidad de los docentes enseñan la forma de como ellos consideran que es el examen, más no enseñan por la experiencia adquirida al presentar multiples veces el examen de calidad, ... es mejor enseñar a recorrer el camino cuando éste ya se ha transitado varias veces, a enseñar a recorrer el camino cuando éste nunca se ha transitado.

Al año 2010 el autor ha presentado seis veces el ECAES en Ingeniería de Sistemas como estrategia para detectar la orientación que año tras año esta tomando el examen. Adicionalmente, después de la presentación de los exámenes ha reescrito las preguntas que han salido en matemáticas computacionales y las ha incorporado a sus guías de

clase para indicarle a sus estudiantes como algunos temas están siendo evaluados en el examen de calidad.

Algunas preguntas reescritas por el autor en la temática de suma de términos de una serie geométrica son las siguientes:

1. La figura muestra un cuadrado de lado 1, el cual se divide en cuatro partes iguales y se somborean dos, luego se toma uno de los cuadrados de lado 0.5, se divide en cuatro partes iguales y se somborean dos, el proceso sigue así sucesivamente¹.



El área de la figura sombreada esta dada por:

- A.) $3/4$
 - B.) $3/5$
 - C.) $1/2$
 - D.) $2/3$
 - E.) $5/7$
2. El perímetro de la figura sombreada en el ejercicio 1 esta dado por:
 - A.) 3.0
 - B.) 3.25
 - C.) 3.5
 - D.) 3.75
 - E.) 4.0
 3. Desde una altura de un metro se deja caer una pelota que rebota medio metro, luego rebota un cuarto de metro, luego rebota un octavo de metro y así sucesivamente hasta que se detiene².

¹Las preguntas 1 y 2 fueron reescritas a partir de preguntas no liberadas del ECAES de Ingeniería de Sistemas del año 2007.

²Pregunta reescrita a partir de una pregunta no liberada del ECAES de Ingeniería de Sistemas del año 2008.

¿Cuál es la suma total en metros de la trayectoria que ha recorrido la pelota hasta que se detiene?.

- A.) 2
- B.) 2.5
- C.) 3
- D.) 3.5
- E.) 47

Al conocer de primera mano la orientación que está tomando el examen de calidad, la forma de como se están evaluando algunos temas y la forma como se deben construir las preguntas según el ICFES³, entonces se pueden entrar a proponer preguntas tipo ECAES como las siguientes:

1. Determine el valor generado por la expresión⁴

$$\lceil [3,3 * \lceil \sqrt{10} \rceil] + 0,01 \rceil + \lceil 0,000001 \rceil$$

- A.) 11
- B.) 12
- C.) 14
- D.) 15
- E.) 16

2. Seleccione la formula que genera la secuencia de enteros: 1, 2, 2, 2, 3, 3, 3, 3, 3, 4, 4, 4, 4, 4, 4, 4, ...

- A.) $f(n) = \lceil \sqrt{n} \rceil + 1, n \geq 0$
- B.) $f(n) = \lfloor \sqrt{n} \rfloor + 1, n \geq 1$
- C.) $f(n) = \lceil \sqrt{n+2} \rceil, n \geq 1$
- D.) $f(n) = \lfloor \sqrt{n-1} \rfloor, n \geq 1$
- E.) $f(n) = \lceil \sqrt{n} \rceil, n \geq 1$

3. Seleccione el resultado de la sumatoria: $5^3 + 5^4 + 5^5 + \dots + 5^n$

- A.) $\frac{5^3(5^{n-2} - 1)}{4}$
- B.) $\frac{5^3(5^{n+1} - 1)}{4}$

³Anexo A: Manual de construcción de preguntas tipo ECAES.

⁴Las preguntas desde la 1 hasta la 5 son del autor.

- C.) $\frac{5^{n+1} - 1}{4}$
D.) $\frac{5^3(5^n - 1)}{4}$
E.) $5^3(5^{n-2} - 1)$

4. El resultado de la sumatoria: $3^0 - 3^1 + 3^2 - 3^3 + 3^4 - 3^5 + \dots + (-1)^n * 3^n$ es:

- A.) $\frac{(-3)^{n+1} - 1}{2}$
B.) $\frac{(-3)^{n+1} + 1}{4}$
C.) $\frac{(-3)^{n+1} + 1}{2}$
D.) $\frac{(-1)^n(3)^{n+1} + 1}{4}$
E.) $\frac{(-1)^{n+1}(3)^{n+1} + 1}{4}$

5. Identifique el resultado de la suma de términos: $1(3) + 2(4) + 3(5) + 4(6) + \dots + n(n+2)$

- A.) $\frac{n(n+2)(2n+1)}{3}$
B.) $\frac{n(n+1)(2n+7)}{6}$
C.) $\frac{n(n+1)(2n-1)}{3}$
D.) $\frac{n(n+1)(n+2)}{3}$
E.) $\frac{n(n+1)(n+2)}{6}$

En el documento anexo, “Material de apoyo del primer curso de matemáticas computacionales”, al final de cada capítulo se encuentra el banco de preguntas tipo ECAES correspondientes a los temas tratados en él. Banco de preguntas construido y recopilado por el autor de ésta investigación.

Capítulo 6

Resultados esperados

- Que el trabajo que se está proponiendo en ésta investigación sea aceptado en la nueva reforma curricular que se está adelantando en el programa de Ingeniería de Sistemas y Computación de la Universidad Tecnológica de Pereira.
- Que el material de apoyo en Matemáticas Computacionales I sea utilizado como material guía en todos los cursos de Matemáticas Computacionales I en el programa de Ingeniería de Sistemas y Computación de la Universidad Tecnológica de Pereira.
- Que el material de apoyo generado en esta investigación sea “liberado” (almacenado) en la página oficial institucional de la asignatura Matemáticas Computacionales I, del programa de Ingeniería de Sistemas y Computación de la Universidad Tecnológica de Pereira.
- Que el material de apoyo generado en esta investigación no sea únicamente conocido y utilizado en la Universidad Tecnológica de Pereira.
- Que el material de apoyo generado en esta investigación sea periódicamente actualizado mejorándolo o corrigiéndolo gracias a la retroalimentación que brinden los profesores o estudiantes que trabajen con el.
- Que este proyecto sea tomado como proyecto piloto por parte del programa de Ingeniería de Sistemas y Computación de la Universidad Tecnológica de Pereira, para la construcción de materiales de apoyo que después sean almacenados en las páginas oficiales institucionales en la universidad, para que estos puedan ser libremente descargados por cualquier persona dentro o fuera de la universidad.

Capítulo 7

Conclusiones

- A partir de la investigación se encontró que se necesita una reforma sustancial con respecto a los cursos de Matemáticas Computacionales en el programa de Ingeniería de Sistemas y Computación de la Universidad Tecnológica de Pereira.
- Con la propuesta de los cursos de Matemáticas Computacionales se espera que los estudiantes tengan la fundamentación adecuada para trabajar en el área de Inteligencia Artificial, Ingeniería de Software, entre otras.
- Algunos de los contenidos de los cursos de Matemáticas Computacionales ya están bien definidos y en la mayoría de universidades del mundo, en su enseñanza se aplican metodologías clásicas, por este motivo este trabajo respeta dicha metodología y le da prioridad al contenido del primer curso de Matemáticas Computacionales, dando libertad de cátedra a los docentes (que orienten esta asignatura) con respecto a la forma de enseñanza, siempre y cuando se respete el contenido del curso.
- La propuesta de programas para las asignaturas del área de Matemáticas Computacionales cumple tanto los lineamientos de Computing Curricula 2005 [ACMIEEE2005] para un programa de Ingeniería de Sistemas en la línea de Ciencias de la Computación a nivel internacional, como también, los lineamientos para un programa de Ingeniería de Sistemas en Colombia según en Ministerio de Educación Nacional [AI2003].
- Este trabajo es una prueba evidente de como los programas de maestría de la institución apoyan y fortalecen académicamente a los programas de pregrado en sus reformas curriculares. El trabajo en equipo entre maestrías y pregrados suma para que la Universidad Tecnológica de Pereira siga siendo acreditada de alta calidad institucional.
- El material de apoyo generado en este proyecto de maestría para el primer curso de Matemáticas Computacionales, esta siendo utilizado con éxito, desde el semestre Febrero-Junio del año 2010 como texto guía de Matemáticas Discretas en el programa de Ingeniería de Sistemas y Computación de la Universidad Tecnológica de Pereira.

- El banco de preguntas tipo ECAES que se construyó para el primer curso de Matemáticas Computacionales cumple con el estándar exigido por el ICFES para éste tipo de instrumentos de evaluación [GEES2006]. Adicionalmente, éste banco de preguntas es vigente y pertinente para los nuevos exámenes SABER PRO.
- Gran parte de la investigación y del material generado en éste trabajo de tesis de maestría ha sido utilizado con éxito durante los últimos tres años, en la preparación de los estudiantes de Ingeniería de Sistemas y Computación de la institución para la presentación del examen de calidad de la educación superior (ECAES). Teniendo solo en cuenta los resultados del ECAES, éstos, no solo han permitido posicionar al programa como el mejor de Ingeniería en UTP, sino también como el mejor programa académico de toda la institución en los años 2008 y 2010. Los resultados del ECAES no solo han permitido posicionar al programa localmente en la institución, también han permitido posicionar al programa a nivel nacional, donde en el año 2008 éste ocupó el octavo puesto entre 119 programas de Ingeniería de Sistemas evaluados en toda Colombia. Pero, el reconocimiento no es solo como programa, también se ha logrado tener estudiantes entre los diez mejores ECAES, en el año 2008 Luis Felipe Abad Guzmán obtuvo un puntaje de 140.94 que le permitió ocupar el sexto puesto a nivel nacional, ahora en los últimos ECAES realizados en el año 2010, Sebastián Gómez Gonzalez ha obtenido un puntaje de 148.25. Puntaje que estadísticamente debería darle el primer puesto a nivel nacional en nuestra carrera.
- Éste no es sólo un trabajo para cumplir un requisito de grado de maestría, es un trabajo para solucionar una problemática real y actual del programa de Ingeniería de Sistemas y Computación de la institución. Es un trabajo que se realizó para utilizarse, porque de lo contrario, serían tres o más años perdidos por parte del autor en su realización.

Capítulo 8

Sugerencias y trabajos futuros

Seguir desarrollando el material de apoyo para la enseñanza del primer curso de Matemáticas Computacionales. Material sugerido el cual debe estar disponible en un sitio de internet¹ :

- Clases filmadas de cada uno de los temas.
- Software para auto evaluación.
- En general utilizar las Tecnologías de la Información con el apoyo de expertos en su uso.

¹Esto forma parte del plan de mejoramiento del programa de Ingeniería de Sistemas de la Universidad Tecnológica de Pereira.

Bibliografía

- [A2005] Acofi. 2005. “Marco de Fundamentación Conceptual: Especificaciones de Prueba ECAES Ingeniería de Sistemas Versión 6” [online]. Bogotá D. C.. Disponible de internet:
http://200.14.205.63:8080/portalicfes/home_2/rec/arc_4400.pdf
- [ACMIEEE2005] ACM, IEEE-CS. 2005. “Computing Curricula 2005, The Overview Report covering undergraduate degree programs in: Computer Engineering, Computer Science, Information Systems, Information Technology and Software Engineering”. A cooperative project of The Association for Computing Machinery (ACM), The Association for Information Systems (AIS) and The Computer Society (IEEE-CS).
Disponible de internet:
http://www.acm.org/education/curric_vols/CC2005-March06Final.pdf
- [AI2003] Acofi - Icfes. 2003. “Exámenes de Estado de Calidad de la Educación Superior (ECAES) en Ingeniería - 2003: Especificaciones de los Exámenes de Estado de Calidad de la Educación en Ingeniería de Sistemas/Informática - 2003” [online]. ICFES, Bogotá D. C..
Disponible de internet:
http://200.14.205.63:8080/portalicfes/home_2/rec/arc_3529.pdf
- [B2002] Biggs, Norman L.. 2002. “Discrete Mathematics”, Second Edition. Oxford University Press.
- [B1988] Bustamente Arias, Alfonso. 1988. “Elementos de Algebra en Ciencias de la Computación”. Serie de textos universitarios, Universidad ICESI, Cali, Colombia.
- [B2009] Bustamente Arias, Alfonso. 2009. “Lógica y Argumentación: De los argumentos inductivos a las álgebras de Boole”. Pearson Educación. Colombia.
- [C1989] Caicedo, Xavier. 1989. “Elementos de Lógica Matemática y Calculabilidad”. Universidad de los Andes, Bogotá, Colombia.
- [C1993] Cardoso, Rodrigo. 1993. “Verificación y Desarrollo de Programas”. Universidad de los Andes, Bogotá, Colombia.

- [CLRS2001] Cormen, T. H., Leiserson, C. E., Rivest, R. L., and Stein, C.. 2001. "Introduction to Algorithms", second edition. The MIT Press, Cambridge, Massachusetts.
- [dC2004] de Castro Korgi, Rodrigo. 2004. "Teoría de la Computación: lenguajes, autómatas, gramáticas", Colección Notas de Clase, Universidad Nacional de Colombia, Facultad de Ciencias, Departamento de Matemáticas, Bogotá, Colombia.
- [dC2003] de Castro Korgi, Rodrigo. 2003. "El Universo L^AT_EX". Segunda edición. Universidad Nacional de Colombia, Facultad de Ciencias, Departamento de Matemáticas, Bogotá, Colombia.
- [DPV2006] Dasgupta, S.. Papadimitriou, C. H.. and Vazirani, U. V.. 2006. "Algorithms". Mc Graw Hill Science/Engineering/Math.
- Disponible en la página del autor:
<http://www.cs.berkeley.edu/~vazirani/algorithms.html>
- [G2003] Grimaldi, Ralph. 2003. "Discrete and Combinatorial Mathematics", Five Edition. Pearson.
- [GEES2006] Grupo Evaluación de la Educación Superior. 2006. "Manual de Construcción de Preguntas para Ingeniería". ICFES.
- [GS2002] Gómez Marín, Raúl y Sicard Ramírez, Andrés. 2002. "Informática Teórica. Elementos Propedéuticos". Fondo Editorial de la Universidad EAFIT, Medellín, Colombia.
- Disponible en la página del autor:
<http://www1.eafit.edu.co/asicard/pubs/informatica-teorica.pdf>
- [GV1998] Guerequeta, Rosa y Vallecillo, Antonio. 1998. "Técnicas de Diseño de Algoritmos". Servicio de Publicaciones de la Universidad de Málaga, España.
- Disponible en la página del autor:
<http://www.lcc.uma.es/~av/Libro/>
- [HMU2002] Hopcroft, J. E.. Motwani, R.. y Ullman, J. D.. 2002. "Introducción a la Teoría de Autómatas, Lenguajes y Computación", Segunda Edición. Pearson Educación S.A.
- [I1997] Isasi Viñuela, Pedro et al. 1997. "Lenguajes Gramáticas y Autómatas un Enfoque Práctico". Addison-Wesley Iberoamericana España.
- [J2008] Johnsonbaugh, Richard. 2008. "Discrete Mathematics". Seven Edition. Prentice Hall.
- [J2009] Jiménez Murillo, José A.. 2009. "Matemáticas para la Computación". Alfaomega.

- [K1995] Kelly, Dean. 1995. “Teoría de Autómatas y Lenguajes Formales”. Prentice Hall.
- [KBR2008] Kolman, Bernard, Busby, Robert y Ross, Sharon C.. 2008. “Estructuras de Matemáticas Discretas para la Computación”, Sexta edición. Prentice Hall.
- [PR2003] Presidencia de la Republica. 2003. “Decreto 1781 de 2003: Por el cual se reglamentan los Exámenes de Estado de Calidad de la Educación Superior, ECAES, de los estudiantes de los programas académicos de pregrado” [online]. Ministerio de Educación Nacional. Bogotá D. C..
Disponible de internet:
http://www.mineducacion.gov.co/normas/descarga/Decreto_1781_2003.pdf
- [R2007] Rosen, Kenneth. H.. 2007. “Discrete Mathematics and Its Applications”, Six edition. Mc Graw Hill.
- [RW2002] Ross, Kenneth A. y Wright, Charles R.. 2002. “Matemáticas Discretas”, Quinta edición. Prentice Hall.
- [S1983] Sedgewick, Robert. 1983. “Algorithms”. Addison Wesley.
- [S2002a] Sedgewick, Robert. 2002. “Algorithms in Java: Parts 1-4: Fundamentals, Data Structures, Sorting and Searching”, Third Edition. Addison Wesley.
- [S2002b] Sedgewick, Robert. 2002. “Algorithms in Java: Parts 5: Graph Algorithms”, Third Edition. Addison Wesley.
- [S1996] Sipser, M.. 1996. “Introduction to the Theory of Computation”. Thomson/Brooks Cole.
- [S2007] Sierra Aristizabal, Manuel. 2007. “Argumentación Deductiva con Diagramas y Árboles de Forzamiento”, Primera edición, MS-Print, Medellín, Colombia.
Disponible en la página:
<http://www1.eafit.edu.co/asicard/teaching/logic-CB0260/SierraAristizabal-2007.Argumentancion-deductiva-con-diagramas-y-arboles-de-forzamiento.pdf>
- [Y2001] Yriarte, Vicente. 2001. “Elementos de Teoría Axiomática de Conjuntos”. Universidad Simón Bolívar. Caracas, Venezuela.
Disponible en la página:
<http://www ldc.usb.ve/%7Emeza/ci-7521/LibroTeoConjYriarte.ps>
- [Y1996] Yriarte, Vicente. 1996. “Elementos de Teoría Combinatoria”. Universidad Simón Bolívar. Caracas, Venezuela.
Disponible en la página del autor:
<http://www ldc.usb.ve/~yriarte/d3index.html>

Apéndice A

Manual de construcción de preguntas tipo ECAES

Esta capítulo anexo ha sido tomado complementamente del “Manual de Construcción de Preguntas Tipo ECAES para Ingeniería”, documento generado por el Grupo de Evaluación de la Educación Superior del Instituto Colombiano para el Fomento de la Educación Superior (ICFES), material referenciado en la bibliografía como [GEES2006].

A.1. Aspectos técnicos

Para que un examen sea técnicamente aceptado debe cumplir con ciertas características como:

Validez: Responde a la pregunta de si el examen mide lo que realmente pretende medir, de modo que sus resultados sean pertinentes. Para ello se debe tener en cuenta el qué y el evaluar, de tal manera que se sustenten adecuadamente las inferencias que cómo se harán a partir de los resultados de la evaluación.

Confiabilidad: Se refiere al grado en que las preguntas de una prueba son consistentes y miden con precisión el objeto de evaluación.

Objetividad: Se refiere a los procedimientos sistemáticos, homogéneos y organizados que soportan los diferentes procedimientos de construcción, validación, aplicación y calificación de la prueba.

A.2. Qué y cómo se evalúa

Competencias

Interpretativa: se relaciona con las acciones para identificar y comprender los elementos y variables que definen un problema o que están implicadas en un fenómeno o proceso; en ingeniería aludiría a las acciones encaminadas a encontrar el sentido de un problema, una gráfica, un plano, un diagrama de flujo, una ecuación, un fenómeno,

entre otras situaciones, referidas al objeto de estudio de la Ingeniería.

Argumentativa: establece las relaciones entre las variables, el análisis de éstas, así como también la explicación y el dar razón sustentada del contexto relacionado con los tópicos de la ingeniería.

Propositiva: hace referencia al planteamiento de hipótesis y a la posibilidad de proponer distintas alternativas de solución a problemas relacionados con la ingeniería.

Componentes

Para la construcción del ECAES de ingeniería la concepción que se ha adoptado de componente es la que corresponde a un escenario en donde típicamente un estudiante al terminar su carrera de ingeniería puede mostrar la formación que ha recibido durante la misma. Con base en esta connotación los componentes adoptados son, entonces, por su enunciado, de carácter transversal. Esto es, son comunes a los diferentes programas.

En los siguientes párrafos, tomados de los Marcos de Fundamentación Conceptual se expresa el significado de tres componentes genéricos propuestos para todas las pruebas de ingeniería.

- **Modelamiento de fenómenos y procesos:** entendido como la concepción de esquemas teóricos, generalmente en forma matemática, de un sistema o de una realidad compleja, que se elabora para facilitar su comprensión, análisis, aplicación y estudio de su comportamiento.
- **Resolución de problemas de ingeniería:** se entiende como soluciones referidas a cualquier situación significativa, desde elementos dados hasta elementos desconocidos, sean éstos reales o hipotéticos. Requiere pensamiento reflexivo y un razonamiento de acuerdo con un conjunto de definiciones, axiomas y reglas. Se pretende lograr este componente a través de las ciencias básicas y con ello tener una fundamentación conceptual muy sólida en matemáticas y física. Esto genera una estructura de pensamiento lógico y simbólico y da las herramientas básicas para la innovación y el desarrollo tecnológico.
- **Diseñar, gestionar y evaluar:** se expresa como la dimensión resultante del análisis y el cálculo; es encontrar las correctas proporciones y las soluciones económicas; es determinar características, aplicar sistemas y procesos que permitan encontrar las óptimas alternativas; es lograr el mejor aprovechamiento de los materiales buscando su sostenibilidad y preservación del medio ambiente; es estimar, apreciar y calcular el valor de algo; es llevar a cabo las acciones y efectos derivados de administrar con el propósito de lograr los objetivos propuestos, entre otros.

Nivel de complejidad de las preguntas

El objetivo del examen no es el de identificar solamente a los mejores estudiantes (para lo cual una prueba con un alto nivel de dificultad sería lo apropiado), sino el de evaluarlos a todos y por este medio, aportar elementos para que las Instituciones de Educación Superior evalúen sus programas. Es por esto que la prueba debe incluir preguntas con niveles de complejidad bajo, medio y alto; ¿la pregunta entonces es dónde se encuentran las cotas superior e inferior en este espectro de complejidad?. Un criterio que podría ayudar consiste en definir lo que se entiende por preguntas “muy difíciles” y preguntas “muy fáciles”, las cuales deben ser descartadas al elaborar la prueba.

Pregunta muy difícil: Aquella que se espera que sólo un 5 % de los estudiantes conteste correctamente.

Pregunta muy fácil: Aquella que se espera que un 95 % o más de los estudiantes conteste correctamente.

Toda pregunta que se encuentre en el rango intermedio es aceptable desde el punto de vista de la complejidad, y se puede clasificar en una de tres categorías:

- Pregunta de complejidad baja: la contestarían correctamente entre un 70 % a un 95 % de los estudiantes.
- Pregunta de complejidad media: la contestarían correctamente entre un 30 % a un 70 % de los estudiantes.
- Pregunta de complejidad alta: la contestarían correctamente entre un 5 % a un 30 % de los estudiantes.

Preguntas de selección múltiple con única respuesta

Los ECAES de ingeniería están conformados por preguntas de selección múltiple con única respuesta, las cuales constan de un enunciado y de cuatro posibilidades de respuesta, entre las cuales se debe escoger la que considere correcta.

Estructura del ítem o pregunta

Contexto: descripción de una situación problema, caso, evento o gráfica en la que se presenta la información suficiente y necesaria para dar solución a las preguntas que se derivan de ella.

Enunciado: parte del ítem en la que se enuncia el problema o tarea que se debe resolver de acuerdo con el contexto planteado.

Opciones de respuesta: lo componen las opciones de respuesta plausibles que guardan relación con el enunciado y el contexto, de las cuales sólo una es correcta (se conoce usualmente como “clave”).

A.3. Reglas para construir preguntas

Se trata de reglas sencillas, pero, difíciles de lograr en la práctica. Como reglas generales básicas se pueden señalar:

- Estudiar y apropiarse del Marco de Fundamentación Conceptual y especificaciones del ECAES de Ingeniería de Sistemas en el cual se va a trabajar, antes de comenzar a elaborar las preguntas.
- Construir en la competencia, el componente y el contenido referencial respectivo, aquellas preguntas que a su criterio no pueden faltar en un ECAES para Ingeniería de Sistemas, dada su importancia como concepto medular de la disciplina o área en cuestión, o bien por su importancia para el ejercicio profesional. Esto es lo que se conoce como pertinencia de la pregunta.
- Preguntarse siempre ¿Qué evalúa este ítem o pregunta?
- Tener en cuenta que a partir de un mismo texto o enunciado o situación problema se pueden derivar varias preguntas.

De acuerdo con lo anterior, para contribuir a que una prueba sea confiable y válida, sus ítems deben estar fielmente interrelacionados con las especificaciones de prueba y con los propósitos de la evaluación. Adicionalmente es importante que su construcción se realice teniendo en cuenta las siguientes recomendaciones:

1. Los enunciados deben ser, preferiblemente, afirmativos, por tanto, se recomienda evitar los enunciados negativos. En caso de ser necesarios, se debe resaltar el **NO** o el **NUNCA** o el **NADIE** para llamar la atención hacia la formulación negativa. La doble negación afecta la comprensión de los evaluados (“No es cierto que no procedan los recursos”).
2. No se deben repetir palabras o expresiones en todas las opciones de respuesta, si éstas se pueden incluir en el enunciado general de la pregunta.
3. Deben evitarse en las opciones las expresiones “todas o ninguna de las anteriores”. En su lugar es necesario construir alternativas de respuesta que sean plausibles y atractivas para las personas que no tengan la competencia o dominio conceptual que exige la pregunta.
4. Evitar expresiones o palabras rebuscadas que puedan confundir al evaluado. Se recomienda emplear un lenguaje directo, preciso, sencillo y comprensible, para evitar preguntas que sean confusas o incoherentes. El vocabulario debe ser adecuado a las características del grupo que presentará la prueba y al área que se pretende evaluar.
5. Las opciones de una pregunta no deben dar indicaciones sobre la clave por ofrecer un cierto contraste evidente:

- De longitud
 - De precisión/imprecisión
 - De uso común/técnico
 - De generalización/particularización, etc.
6. Todas las preguntas de una prueba deben ser independientes entre sí. La información de una pregunta no debe servir de pauta para contestar otra, ni la respuesta a una pregunta debe depender de haber encontrado primero la de otra anterior. Pero sí se pueden construir varias preguntas con un enunciado o situación común.
 7. Evitar enunciados demasiado extensos y poco atractivos ya que desmotivan la lectura, disminuyen el tiempo de respuesta y fatigan al estudiante. Es necesario calibrar adecuadamente su extensión y precisar la información que es imprescindible que contengan.
 8. Evitar los items que pueden contestarse por mera lógica o sentido común y aquellos cuya respuesta dependa únicamente de recordar un término, un símbolo, un dato o la fecha en que ocurrió un evento.
 9. Verificar que la pregunta corresponda con la estructura de la prueba y con los propósitos de la evaluación.
 10. No conducir a juicios de valor. Cada pregunta debe poseer una base terminológica, conceptual y disciplinar que evite caer en afirmaciones imprecisas o ambiguas, las cuales frecuentemente llevan a establecer juicios de valor por parte del examinado.
 11. La ubicación de las claves en la prueba no debe seguir ninguna secuencia. Deben colocarse al azar.
 12. No debe utilizarse en las opciones las mismas palabras significativas que se utilizaron en el enunciado. Repetir la misma palabra del enunciado en cualquiera de las opciones lleva a elegirla como respuesta, sin serlo necesariamente.
 13. El número de opciones debe ser constante para toda la sección o parte de la prueba.

MATERIAL DE APOYO PARA EL PRIMER CURSO DE MATEMÁTICAS COMPUTACIONALES.

Ing. HUGO HUMBERTO MORALES PEÑA

MAESTRÍA EN ENSEÑANZA DE LAS MATEMÁTICAS
Línea de Matemáticas Computacionales

UNIVERSIDAD TECNOLÓGICA DE PEREIRA
FACULTAD DE CIENCIAS BÁSICAS
DEPARTAMENTO DE MATEMÁTICAS

Pereira, Risaralda
25 de Febrero de 2011

Índice general

1. Introducción a la lógica matemática	7
1.1. Cálculo proposicional	7
1.1.1. Conectivos proposicionales	8
1.1.2. Fórmulas bien formadas	9
1.1.3. Fórmulas lógicamente equivalentes (FLE)	11
1.1.4. Tautología	11
1.1.5. Leyes de la lógica	13
1.1.6. Utilizando las leyes de la lógica proposicional	14
1.1.7. Conectivo X-OR (\otimes)	15
1.1.8. Conectivo NOR (\downarrow)	17
1.1.9. Conectivo NAND (\uparrow)	20
1.1.10. Ejercicios	22
1.2. Reglas de inferencia	25
1.2.1. Tabla de reglas de inferencia	25
1.2.2. Utilización de las reglas de inferencia para demostrar la validez de razonamientos	29
1.2.3. Ejercicios	34
1.3. Lógica de predicados	37
1.3.1. Cuantificador universal	38
1.3.2. Cuantificador existencial	40
1.3.3. Variables ligadas	42
1.3.4. Alcance de un cuantificador	42
1.3.5. Negaciones y cuantificadores	43
1.3.6. Ejercicios	44
1.4. Preguntas Tipo ECAES	48
2. Uso de sucesiones y sumatorias en computación	53
2.1. Funciones piso y techo	53
2.1.1. Propiedades de las funciones piso y techo	53
2.2. Sucesiones	55
2.3. Sucesiones especiales de números	57
2.4. Sumatorias	61
2.4.1. Fórmulas de sumatorias útiles:	68
2.5. Sumatorias dobles y su utilización en ciclos anidados	70
2.6. Notaciones computacionales (O, Omega, Theta)	74

2.6.1.	Notación O	74
2.6.2.	Notación Ω	76
2.6.3.	Notación Θ	78
2.7.	Complejidad de algoritmos iterativos	79
2.7.1.	El Algoritmo de Ordenamiento por Selección	79
2.7.2.	El Algoritmo de Ordenamiento por Inserción	80
2.8.	La programación recursiva y su costo computacional	82
2.8.1.	La sucesión de Fibonacci	83
2.8.2.	Fibonacci mejorado	85
2.9.	Ejercicios	86
2.10.	Preguntas tipo ECAES	90
3.	Técnicas de demostración	101
3.1.	Técnica de demostración directa.	101
3.2.	Técnica de demostración indirecta	103
3.2.1.	Técnica de demostración por contra-recíproca	103
3.2.2.	Técnica de demostración por contradicción.	105
3.3.	Técnica de demostración por disyunción de casos	108
3.4.	Técnica de demostración por contraejemplo	114
3.5.	Técnica de demostración por inducción matemática	117
3.6.	Ejercicios	131
3.7.	Preguntas tipo ECAES	134
4.	Relaciones de recurrencia	137
4.1.	Método de Iteración	138
4.2.	Complejidad de algoritmos recursivos	152
4.2.1.	El Algoritmo de Ordenamiento por Montones	154
4.3.	Preguntas tipo ECAES	161
5.	Conjuntos	171
5.1.	El conjunto potencia	174
5.2.	Producto cartesiano	174
5.3.	Operaciones de conjuntos	176
5.4.	Identidades en conjuntos	179
5.5.	Uniones e intersecciones generalizadas	181
5.6.	Ejercicios	181
5.7.	Preguntas tipo ECAES	184
6.	Funciones	187
6.1.	Conceptos fundamentales	187
6.2.	Funciones inyectivas (o funciones uno a uno)	189
6.3.	Funciones sobreyectivas	189
6.4.	Funciones biyectivas	190
6.5.	Funciones inversas y composición de funciones	190
6.6.	Gráfica de una función	193

6.7. Ejercicios	194
6.8. Preguntas tipo ECAES	197
7. Relaciones	207
7.1. Relación binaria	207
7.2. Funciones como relaciones	208
7.3. Relaciones en un conjunto	208
7.4. Propiedades de las relaciones	210
7.4.1. Propiedad de reflexividad	210
7.4.2. Propiedad de simetría	211
7.4.3. Propiedad de antisimetría	211
7.4.4. Propiedad de transitividad	211
7.5. Combinación de relaciones	213
7.6. Composición y potencia de relaciones	213
7.7. Representación de relaciones	214
7.7.1. Representación de relaciones utilizando matrices	214
7.8. Relaciones n-arias y su uso computacional	218
7.9. El álgebra relacional en bases de datos	218
7.9.1. La operación de selección	219
7.9.2. La operación proyección	220
7.9.3. Composición de operaciones relacionales	221
7.9.4. La operación de unión	222
7.9.5. La operación de diferencia	223
7.9.6. La operación de intersección	223
7.9.7. La operación de asignación	224
7.9.8. La operación producto cartesiano	224
7.10. SQL el lenguaje de consulta de las bases de datos	227
7.10.1. La cláusula SELECT	228
7.10.2. La cláusula FROM	229
7.10.3. La cláusula WHERE	229
7.10.4. Operaciones sobre relaciones	230
7.10.5. La operación de unión	230
7.10.6. La operación de diferencia	230
7.10.7. La operación de intersección	231
7.11. Ejercicios	231
7.12. Preguntas tipo ECAES	234
8. Relaciones de equivalencia	237
8.1. Clases de equivalencia	239
8.2. Clases de equivalencia y particiones	239
8.3. Conjuntos parcialmente ordenados	241
8.4. Ejercicios	245
8.5. Preguntas tipo ECAES	247

9. Introducción a la teoría de números	249
9.1. Los números enteros y la división	249
9.1.1. Introducción	249
9.1.2. División entre números enteros	249
9.1.3. El algoritmo de la división entre números enteros	251
9.1.4. Los números primos	251
9.1.5. Teorema fundamental de la aritmética	252
9.1.6. Procedimiento para generar la factorización prima de un número entero	256
9.1.7. El máximo común divisor (MCD)	260
9.1.8. El mínimo común múltiplo(MCM)	261
9.1.9. El algoritmo de Euclides	262
9.2. Aritmética modular	263
9.2.1. Aplicaciones de la aritmética modular	264
9.2.2. Asignación de localizaciones de memoria en el computador . . .	264
9.2.3. Generación de números pseudoaleatorios	265
9.2.4. Criptosistemas basados en aritmética modular	266
9.3. Representación de los enteros en el computador	267
9.3.1. Representación de números enteros en base hexadecimal	269
9.3.2. Cambio de base de un número entero escrito en base 10	269
9.3.3. Algoritmo para construir la expansión de n en base b	270
9.3.4. Algoritmos para operaciones de números enteros en base 2 . . .	272
9.4. Ejercicios	277
9.5. Preguntas tipo ECAES	281

Capítulo 1

Introducción a la lógica matemática

1.1. Cálculo proposicional

Definición de proposición:

Una proposición es un enunciado declarativo que puede ser calificado sin ambigüedad como verdadero o falso. En este análisis no se tendrán en cuenta proposiciones que requieran una opinión individual y que por lo tanto, no pueden ser verdaderas o falsas.

Las siguientes declaraciones son ejemplos de proposiciones:

- Matemáticas Discretas es una materia que se evalúa en el Examen de Calidad de la Educación Superior (ECAES) en el programa académico de Ingeniería de Sistemas
- El promedio a nivel nacional en el ECAES de Ingeniería de Sistemas fue de 110.3 puntos en el año 2007
- El cuadernillo de inglés tenía 40 preguntas en el ECAES de Ingeniería de Sistemas del año 2008

Las siguientes declaraciones son ejemplos de lo que no es una proposición:

- Atlético Nacional es el mejor equipo del fútbol colombiano
- Álvaro Uribe ha sido el mejor presidente de los colombianos
- Los próximos juegos deportivos nacionales serán ganados por el departamento del Valle

Las proposiciones pueden considerarse como primitivas, ya que en realidad no se pueden descomponer en partes más simples.

Las proposiciones primitivas se utilizan con conectivos lógicos para formar proposiciones compuestas.

Los símbolos p, q, r, s, \dots se utilizarán para denotar proposiciones, los cuales se llamarán variables proposicionales.

1.1.1. Conectivos proposicionales

Los conectivos proposicionales son también conocidos con el nombre de conectivos lógicos.

Los conectivos principales son:

- la negación, representada por el símbolo \sim
- la disyunción, representada por el símbolo \vee
- la conjunción, representada por el símbolo \wedge
- el condicional (o implicación), representada por el símbolo \rightarrow
- el bicondicional (o doble implicación), representada por el símbolo \leftrightarrow

Las tablas de verdad para estos conectivos son:

Tabla de verdad de la negación:

p	$\sim p$
V	F
F	V

Tabla de verdad de la disyunción:

p	q	$p \vee q$
V	V	V
V	F	V
F	V	V
F	F	F

Tabla de verdad de la conjunción:

p	q	$p \wedge q$
V	V	V
V	F	F
F	V	F
F	F	F

Tabla de verdad del condicional:

p	q	$p \rightarrow q$
V	V	V
V	F	F
F	V	V
F	F	V

Tabla de verdad del bicondicional:

p	q	$p \leftrightarrow q$
V	V	V
V	F	F
F	V	F
F	F	V

Dos proposiciones p y q son *equivalentes* cuando el bicondicional $p \leftrightarrow q$ es una proposición verdadera.

Ejemplo 1:

Se tienen las siguientes dos proposiciones:

- p : $\sqrt{2}$ es un número irracional
- q : un año bisiesto tiene 366 días

las dos proposiciones p y q son verdaderas, como $V \leftrightarrow V$ es verdadero, entonces las proposiciones p y q son equivalentes.

Ejemplo 2:

Se tienen las siguientes dos proposiciones:

- p : $2+3=7$
- q : 4 es un número impar

las dos proposiciones p y q son falsas, como $F \leftrightarrow F$ es verdadero, entonces las proposiciones p y q son equivalentes.

1.1.2. Fórmulas bien formadas

Una fórmula es una sucesión finita de variables proposicionales, conectivos lógicos y paréntesis. Una Fórmula Bien Formada (FBF), es, intuitivamente una fórmula coherente, con sentido gramatical. Las FBF serán denotadas por los símbolos: $A, B, C, A_1, B_1, C_1, \dots$

Definición:

Una FBF del cálculo proposicional es aquella fórmula que se ajusta a cualquiera de estos casos:

1. Toda variable proposicional aislada es una FBF
2. Si A es una FBF, entonces $\sim (A)$ es una FBF
3. Si A y B son FBF, entonces también lo son:
 - $(A) \wedge (B)$,
 - $(A) \vee (B)$,
 - $(A) \rightarrow (B)$, y
 - $(A) \leftrightarrow (B)$.
4. Una fórmula es bien formada si lo es como resultado de aplicar los casos 1, 2 y 3 un número finito de veces.

Ejemplo 3:

¿La fórmula $((p) \wedge (\sim (q))) \rightarrow ((\sim (\sim (p))) \leftrightarrow (q))$ es una fórmula bien formada?

Para determinar si la fórmula es bien formada, sea A_0 la fórmula que represente a ésta, si A_0 se puede obtener al aplicar un número finito de pasos los casos 1, 2 y 3 entonces la fórmula es bien formada, para esto se tiene:

$A_0 = ((p) \wedge (\sim (q))) \rightarrow ((\sim (\sim (p))) \leftrightarrow (q))$, como el conectivo principal de la fórmula es la implicación, entonces A_0 se puede representar como $A_0 = (A_1) \rightarrow (B_1)$ donde $A_1 = (p) \wedge (\sim (q))$ y $B_1 = (\sim (\sim (p))) \leftrightarrow (q)$. En el análisis de las fórmulas A_1 y B_1 se tiene:

- $A_1 = (p) \wedge (\sim (q))$, A_1 puede ser reescrita como $A_1 = (A_2) \wedge (B_2)$, con $A_2 = p$ y $B_2 = \sim (q)$, con $B_2 = \sim (B_3)$ y $B_3 = q$. Como A_2 y B_3 son variables proposicionales aisladas entonces éstas son FBF; como B_2 es la negación de una FBF entonces ésta también es una FBF; como la fórmula A_1 es la conjunción de las FBF A_2 y B_2 entonces A_1 es también una FBF.
- $B_1 = (\sim (\sim (p))) \leftrightarrow (q)$, B_1 puede ser reescrita como $B_1 = (A_4) \leftrightarrow (B_4)$, con $A_4 = \sim (\sim (p))$ y $B_4 = q$, como B_4 es una variable proposicional aislada entonces es una FBF. La fórmula A_4 puede ser reescrita como $A_4 = \sim (A_5)$ donde $A_5 = \sim (p)$ que puede ser reescrita como $A_5 = \sim (A_6)$ y $A_6 = p$, como A_6 es una variable proposicional aislada entonces es una FBF; A_5 es la negación de una FBF entonces ésta también es una FBF; A_4 es la negación de A_5 que es una FBF entonces A_4 también es una FBF. Como la fórmula B_1 es la doble implicación entre las FBF A_4 y B_4 entonces B_1 es también una FBF.

en el análisis anterior ya se obtuvo que A_1 y B_1 son FBF y como $A_0 = (A_1) \rightarrow (B_1)$ entonces A_0 es también una FBF.

1.1.3. Fórmulas lógicamente equivalentes (FLE)

Dos fórmulas A y B son lógicamente equivalentes, lo cual se indica en este trabajo como $A \Leftrightarrow B$, cuando tienen la misma tabla de verdad.

Ejemplo 4:

Se tienen las siguientes fórmulas $A = \sim (p \wedge \sim q)$ y $B = \sim p \vee q$, ¿son las fórmulas A y B lógicamente equivalentes?

Para dar respuesta a ésta pregunta se hará uso de las tablas de verdad, para lo cual se tiene:

p	q	$\sim p$	$\sim q$	$p \wedge \sim q$	$\sim (p \wedge \sim q)$	$\sim p \vee q$
V	V	F	F	F	V	V
V	F	F	V	V	F	F
F	V	V	F	F	V	V
F	F	V	V	F	V	V

como las columnas de la tabla que indican los valores para las fórmulas $\sim (p \wedge \sim q)$ y $\sim p \vee q$ son iguales entonces éstas fórmulas son lógicamente equivalentes.

1.1.4. Tautología

Cuando dos fórmulas A y B son lógicamente equivalentes, el bicondicional $A \Leftrightarrow B$ es siempre verdadero. Cuando dos fórmulas A y B son lógicamente equivalentes entonces $A \Leftrightarrow B$ es una tautología, según la definición siguiente:

Definición:

Si una FBF tiene siempre el valor verdadero independientemente de cada asignación particular de valores a sus variables, entonces esta fórmula es una tautología y se denota con V ; si tal valor es siempre falso, entonces esta fórmula es una contradicción y se denota con F .

Ejemplo 5:

¿La fórmula $((p \rightarrow q) \wedge \sim p) \rightarrow (\sim q)$ es una tautología?

Para dar respuesta a esta pregunta se hará uso de las tablas de verdad, para lo cual se tiene:

p	q	$\sim p$	$\sim q$	$p \rightarrow q$	$(p \rightarrow q) \wedge \sim p$	$((p \rightarrow q) \wedge \sim p) \rightarrow (\sim q)$
V	V	F	F	V	F	V
V	F	F	V	F	F	V
F	V	V	F	V	V	F
F	F	V	V	V	V	V

la fórmula no es una tautología porque existe una combinación de asignación de valores de las variables proposicionales que hacen que la fórmula genere el valor falso, dicha asignación de valores es $p = F$ y $q = V$, lo cual se evidencia en la tercer fila de la tabla de verdad.

Una alternativa es indagar de forma indirecta la posibilidad de que alguna combinación de valores dé un valor F en la fórmula.

Manera alternativa:

El conectivo principal de la fórmula es una implicación (\rightarrow), la única posibilidad para que una implicación tome valor falso es cuando el antecedente es verdadero y el consecuente es falso

$$\underbrace{\underbrace{((p \rightarrow q) \wedge (\sim p))}_V}_{F} \rightarrow \underbrace{(\sim q)}_F$$

la única posibilidad para que el consecuente sea falso es cuando la variable proposicional q toma valor verdadero; tener en cuenta que la asignación de valor para la variable proposicional q aplica para toda la expresión

$$\underbrace{\underbrace{((p \rightarrow \underbrace{q}_V) \wedge (\sim p))}_V}_{F} \rightarrow \underbrace{(\sim \underbrace{q}_V)}_F$$

la única posibilidad para que el antecedente sea verdadero es cuando la variable proposicional p toma valor falso; tener en cuenta que la asignación de valor para la variable proposicional p aplica para todas las ocurrencias de dicha variable en la expresión

$$\underbrace{\underbrace{\underbrace{((\underbrace{p}_F \rightarrow \underbrace{q}_V) \wedge (\sim \underbrace{p}_F))}_V}_{V}}_V}_{F} \rightarrow \underbrace{(\sim \underbrace{q}_V)}_F$$

en este ejemplo utilizando la manera alternativa se concluye igualmente que la fórmula no es una tautología porque se logró determinar una asignación de valores para las variables proposicionales que hace que la fórmula tome el valor falso.

1.1.5. Leyes de la lógica

La siguiente tabla contiene las principales leyes de la lógica en el cálculo proposicional, donde A , B y C son fórmulas bien formadas. Para una mayor claridad con respecto al alcance de una negación en un fórmula bien formada en el cálculo proposicional, entonces se usará en la tabla y en el resto de ejercicios de la sección, la representación \overline{A} en vez de $\sim A$, donde A es una fórmula bien formada.

Número	Equivalencia Lógica	Nombre Ley
1.	$\overline{\overline{A}} \Leftrightarrow A$	Ley de doble negación
2.	$\overline{A \vee B} \Leftrightarrow \overline{A} \wedge \overline{B}$	Ley de De Morgan
2'.	$\overline{A \wedge B} \Leftrightarrow \overline{A} \vee \overline{B}$	Ley de De Morgan
3.	$A \vee B \Leftrightarrow B \vee A$	Ley conmutativa
3'.	$A \wedge B \Leftrightarrow B \wedge A$	Ley conmutativa
4.	$A \vee (B \vee C) \Leftrightarrow (A \vee B) \vee C$	Ley asociativa
4'.	$A \wedge (B \wedge C) \Leftrightarrow (A \wedge B) \wedge C$	Ley asociativa
5.	$A \vee (B \wedge C) \Leftrightarrow (A \vee B) \wedge (A \vee C)$	Ley distributiva
5'.	$A \wedge (B \vee C) \Leftrightarrow (A \wedge B) \vee (A \wedge C)$	Ley distributiva
6.	$A \vee A \Leftrightarrow A$	Ley de idempotencia
6'.	$A \wedge A \Leftrightarrow A$	Ley de idempotencia
7.	$A \vee F \Leftrightarrow A$	Ley de identidad
7'.	$A \wedge V \Leftrightarrow A$	Ley de identidad
8.	$A \vee \overline{A} \Leftrightarrow V$	Ley inversa
8'.	$A \wedge \overline{A} \Leftrightarrow F$	Ley inversa
9.	$A \vee V \Leftrightarrow V$	Ley de dominación
9'.	$A \wedge F \Leftrightarrow F$	Ley de dominación
10.	$A \vee (A \wedge B) \Leftrightarrow A$	Ley de absorción
10'.	$A \wedge (A \vee B) \Leftrightarrow A$	Ley de absorción
11.	$(A \rightarrow B) \Leftrightarrow (\overline{B} \rightarrow \overline{A})$	Ley de transposición

1.1.6. Utilizando las leyes de la lógica proposicional

Ejemplo 6:

Simplificar la siguiente proposición compuesta $[(\bar{x} \wedge y) \vee (x \wedge \bar{y})] \rightarrow x$ utilizando las leyes de la lógica proposicional. En cada paso especificar la regla que se utilizó.

Utilizando las leyes de la lógica proposicional la simplificación es la siguiente:

$$\begin{aligned}
 &[(\bar{x} \wedge y) \vee (x \wedge \bar{y})] \rightarrow x \\
 &\iff [(\overline{(\bar{x} \wedge y) \vee (x \wedge \bar{y})})] \vee x, \quad \text{equivalencia lógica de la implicación.} \\
 &\iff [(\overline{\bar{x} \wedge y}) \wedge \overline{x \wedge \bar{y}}] \vee x, \quad \text{ley de De Morgan.} \\
 &\iff [(x \vee \bar{y}) \wedge (\bar{x} \vee y)] \vee x, \quad \text{ley de De Morgan y ley de doble negación.} \\
 &\iff [((x \vee \bar{y}) \wedge \bar{x}) \vee ((x \vee \bar{y}) \wedge y)] \vee x, \quad \text{ley distributiva.} \\
 &\iff [((x \wedge \bar{x}) \vee (\bar{x} \wedge \bar{y})) \vee ((x \wedge y) \vee (\bar{y} \wedge y))] \vee x, \quad \text{ley distributiva.} \\
 &\iff [(F \vee (\bar{x} \wedge \bar{y})) \vee ((x \wedge y) \vee F)] \vee x, \quad \text{ley inversa.} \\
 &\iff [(\bar{x} \wedge \bar{y}) \vee (x \wedge y)] \vee x, \quad \text{ley de identidad.} \\
 &\iff (\bar{x} \wedge \bar{y}) \vee ((x \wedge y) \vee x), \quad \text{ley asociativa.} \\
 &\iff (\bar{x} \wedge \bar{y}) \vee x \quad \text{ley de Absorción} \\
 &\iff (\bar{x} \vee x) \wedge (\bar{y} \vee x), \quad \text{ley distributiva.} \\
 &\iff V \wedge (\bar{y} \vee x), \quad \text{ley inversa} \\
 &\iff (\bar{y} \vee x), \quad \text{ley de identidad.} \\
 &\iff y \rightarrow x, \quad \text{equivalencia lógica de la implicación.}
 \end{aligned}$$

Se puede utilizar tablas de verdad, como mecanismo adicional, para verificar si el resultado obtenido en la simplificación es correcto. Es importante tener en cuenta que la validez del resultado de la simplificación, depende única y exclusivamente, del correcto uso de las leyes de la lógica proposicional, donde, en cada paso de la simplificación se garantiza que se tiene una proposición compuesta lógicamente equivalente a la proposición compuesta original.

x	y	$(\bar{x} \wedge y) \vee (x \wedge \bar{y})$	$[(\bar{x} \wedge y) \vee (x \wedge \bar{y})] \rightarrow x$	$y \rightarrow x$
V	V	F	V	V
V	F	V	V	V
F	V	V	F	F
F	F	F	V	V

Como se obtuvieron exactamente los mismos valores en las columnas de la tabla de

verdad correspondientes a $[(\bar{x} \wedge y) \vee (x \wedge \bar{y})] \rightarrow x$ y a $y \rightarrow x$ entonces las dos proposiciones compuestas son lógicamente equivalentes y la simplificación es correcta.

Ejemplo 7:

Utilizando las leyes de la lógica proposicional simplificar la siguiente proposición compuesta $[(p \vee q) \wedge (p \rightarrow q)] \vee [(\overline{p \vee q}) \wedge (\overline{p \rightarrow q})]$.

Los pasos de la simplificación son los siguientes:

$$\begin{aligned}
 & [(p \vee q) \wedge (p \rightarrow q)] \vee [(\overline{p \vee q}) \wedge (\overline{p \rightarrow q})] \\
 & \iff [(p \vee q) \wedge (\bar{p} \vee q)] \vee [(\overline{p \vee q}) \wedge (\overline{\bar{p} \vee q})], \text{ equivalencia lógica implicación.} \\
 & \iff [(p \vee q) \wedge (\bar{p} \vee q)] \vee [(\bar{p} \wedge \bar{q}) \wedge (p \wedge \bar{q})], \text{ ley de De Morgan.} \\
 & \iff [(p \wedge \bar{p}) \vee q] \vee [(\bar{p} \wedge \bar{q}) \wedge (p \wedge \bar{q})], \text{ ley distributiva.} \\
 & \iff [F \vee q] \vee [(\bar{p} \wedge \bar{q}) \wedge (p \wedge \bar{q})], \text{ ley inversa.} \\
 & \iff [q] \vee [(\bar{p} \wedge \bar{q}) \wedge (p \wedge \bar{q})], \text{ ley de identidad.} \\
 & \iff q \vee [\bar{p} \wedge (\bar{q} \wedge p) \wedge \bar{q}], \text{ ley asociativa.} \\
 & \iff q \vee [\bar{p} \wedge (p \wedge \bar{q}) \wedge \bar{q}], \text{ ley conmutativa.} \\
 & \iff q \vee [(\bar{p} \wedge p) \wedge (\bar{q} \wedge \bar{q})], \text{ ley asociativa.} \\
 & \iff q \vee [F \wedge (\bar{q} \wedge \bar{q})], \text{ ley inversa.} \\
 & \iff q \vee F, \text{ ley de dominación.} \\
 & \iff q, \text{ ley de identidad.}
 \end{aligned}$$

1.1.7. Conectivo X-OR (\otimes)

El conectivo \otimes de la lógica proposicional es llamado O Exclusivo o X-OR. Su tabla de verdad es:

p	q	$p \otimes q$
V	V	F
V	F	V
F	V	V
F	F	F

La siguiente equivalencia lógica representa al X-OR: $p \otimes q \iff (\bar{p} \wedge q) \vee (p \wedge \bar{q})$.

Ejemplo 8:

Determinar sin utilizar tablas de verdad si la proposición compuesta $\overline{(x \otimes y)} \rightarrow (x \vee \bar{y})$ es una tautología.

Haciendo uso de las leyes de la lógica proposicional, el análisis es el siguiente:

$$\begin{aligned}
\overline{(x \otimes y)} \rightarrow (x \vee \overline{y}) &\iff \overline{\overline{(x \otimes y)} \vee (x \vee \overline{y})}, \quad \text{equivalencia lógica de la implicación} \\
&\iff (x \otimes y) \vee (x \vee \overline{y}), \quad \text{ley de doble negación} \\
&\iff ((\overline{x} \wedge y) \vee (x \wedge \overline{y})) \vee (x \vee \overline{y}), \quad \text{equivalencia lógica del X-OR} \\
&\iff (\overline{x} \wedge y) \vee ((x \wedge \overline{y}) \vee (x \vee \overline{y})), \quad \text{ley asociativa} \\
&\iff (\overline{x} \wedge y) \vee (((x \wedge \overline{y}) \vee x) \vee \overline{y}), \quad \text{ley asociativa} \\
&\iff (\overline{x} \wedge y) \vee ((x) \vee \overline{y}), \quad \text{ley de absorción} \\
&\iff (\overline{x} \wedge y) \vee \overline{(x \vee \overline{y})}, \quad \text{ley de doble negación} \\
&\iff (\overline{x} \wedge y) \vee (\overline{x} \wedge \overline{\overline{y}}), \quad \text{ley de De Morgan} \\
&\iff (\overline{x} \wedge y) \vee \overline{(\overline{x} \wedge y)}, \quad \text{ley de doble negación} \\
&\iff V, \quad \text{ley inversa}
\end{aligned}$$

Queda demostrado que la proposición compuesta original es una tautología porque el valor obtenido como resultado de la simplificación es el V .

Ejemplo 9:

Demostrar utilizando las leyes de la lógica proposicional, que la proposición compuesta $\overline{(p \wedge q)} \vee (p \otimes q)$ es lógicamente equivalente a la proposición compuesta $p \wedge q$.

La demostración se justifica con cada uno de los siguientes pasos:

$$\begin{aligned}
\overline{\overline{(p \wedge q)} \vee (p \otimes q)} &\iff \overline{\overline{(p \wedge q)} \wedge \overline{(p \otimes q)}}, \quad \text{ley de De Morgan} \\
&\iff (p \wedge q) \wedge \overline{(p \otimes q)}, \quad \text{ley de doble negación} \\
&\iff (p \wedge q) \wedge \overline{((p \wedge \overline{q}) \vee (\overline{p} \wedge q))}, \quad \text{equivalencia lógica del X-OR} \\
&\iff (p \wedge q) \wedge (\overline{(p \wedge \overline{q})} \wedge \overline{(\overline{p} \wedge q)}), \quad \text{ley de De Morgan} \\
&\iff (p \wedge q) \wedge ((\overline{p} \vee q) \wedge (p \vee \overline{q})), \quad \text{ley de De Morgan} \\
&\iff [(p \wedge q) \wedge (\overline{p} \vee q)] \wedge (p \vee \overline{q}), \quad \text{ley asociativa} \\
&\iff [((p \wedge q) \wedge \overline{p}) \vee ((p \wedge q) \wedge q)] \wedge (p \vee \overline{q}), \quad \text{ley distributiva} \\
&\iff [((q \wedge p) \wedge \overline{p}) \vee ((p \wedge q) \wedge q)] \wedge (p \vee \overline{q}), \quad \text{ley conmutativa} \\
&\iff [(q \wedge (p \wedge \overline{p})) \vee (p \wedge (q \wedge q))] \wedge (p \vee \overline{q}), \quad \text{ley asociativa} \\
&\iff [(q \wedge F) \vee (p \wedge q)] \wedge (p \vee \overline{q}), \quad \text{Leyes inversa y de idempotencia} \\
&\iff [F \vee (p \wedge q)] \wedge (p \vee \overline{q}), \quad \text{ley de dominación}
\end{aligned}$$

$$\begin{aligned}
&\Longleftrightarrow (p \wedge q) \wedge (p \vee \bar{q}), \quad \text{ley de identidad} \\
&\Longleftrightarrow ((p \wedge q) \wedge p) \vee ((p \wedge q) \wedge \bar{q}), \quad \text{ley distributiva} \\
&\Longleftrightarrow (p \wedge (p \wedge q)) \vee ((p \wedge q) \wedge \bar{q}), \quad \text{ley conmutativa} \\
&\Longleftrightarrow ((p \wedge p) \wedge q) \vee (p \wedge (q \wedge \bar{q})), \quad \text{ley asociativa} \\
&\Longleftrightarrow (p \wedge q) \vee (p \wedge F), \quad \text{leyes idempotente e inversa} \\
&\Longleftrightarrow (p \wedge q) \vee F, \quad \text{ley de dominación} \\
&\Longleftrightarrow p \wedge q, \quad \text{ley de identidad}
\end{aligned}$$

1.1.8. Conectivo NOR (\downarrow)

El conectivo NOR es un conectivo completo, en el sentido que, cualquier fórmula del cálculo proposicional puede ser escrita utilizando únicamente éste conectivo.

Mnemotécnicamente:

$$NOR \approx Not\ or \approx No\ o \approx \sim (p \vee q) \approx \overline{(p \vee q)} \approx p \not\vee q \approx p \downarrow q$$

Ejemplo 10:

Demostrar que el conectivo NOR es un conectivo completo.

Sugerencia: Para que el conectivo NOR sea un conectivo completo se debe presentar el equivalente de los siguientes conectivos principales utilizando únicamente el conectivo NOR:

- \sim
- \vee
- \wedge
- \rightarrow
- \leftrightarrow

Representación de la negación utilizando únicamente el NOR:

$$\begin{aligned}
\bar{p} &\Longleftrightarrow \overline{p \vee p}, \quad \text{ley de idempotencia.} \\
&\Longleftrightarrow p \downarrow p, \quad \text{equivalencia lógica NOR.}
\end{aligned}$$

Representación de la disyunción utilizando únicamente el NOR:

$$\begin{aligned}
p \vee q &\Longleftrightarrow (p \vee q) \wedge (p \vee q), \quad \text{ley de idempotencia.} \\
&\Longleftrightarrow \overline{\overline{(p \vee q) \wedge (p \vee q)}}, \quad \text{ley de la doble negación.}
\end{aligned}$$

$$\iff \overline{\overline{(p \vee q)} \vee \overline{(p \vee q)}}, \text{ ley de De Morgan.}$$

$$\iff (p \downarrow q) \downarrow (p \downarrow q), \text{ equivalencia lógica NOR.}$$

Representación de la conjunción utilizando únicamente el NOR:

$$p \wedge q \iff \overline{\overline{(p \wedge q)}}, \text{ ley de doble negación.}$$

$$\iff \overline{\overline{p} \vee \overline{q}}, \text{ ley de De Morgan.}$$

$$\iff \overline{\overline{(p \vee p)} \vee \overline{(q \vee q)}}, \text{ ley de idempotencia.}$$

$$\iff (p \downarrow p) \downarrow (q \downarrow q), \text{ equivalencia lógica NOR.}$$

Representación de la implicación utilizando únicamente el NOR:

$$p \rightarrow q \iff \overline{p} \vee q, \text{ equivalencia lógica de la implicación.}$$

$$\iff (\overline{p} \downarrow q) \downarrow (\overline{p} \downarrow q), \text{ representación de la disyunción con el NOR.}$$

$$\iff ((p \downarrow p) \downarrow q) \downarrow ((p \downarrow p) \downarrow q), \text{ representación de la negación con el NOR.}$$

Representación de la doble implicación utilizando únicamente el NOR:

$$p \leftrightarrow q \iff (p \rightarrow q) \wedge (q \rightarrow p), \text{ equivalencia lógica de la doble implicación.}$$

$$\iff (\overline{p} \vee q) \wedge (\overline{q} \vee p), \text{ equivalencia lógica de la implicación.}$$

$$\iff \overline{\overline{(\overline{p} \vee q) \wedge (\overline{q} \vee p)}}, \text{ ley de la doble negación.}$$

$$\iff \overline{\overline{(\overline{p} \vee q)} \vee \overline{\overline{(\overline{q} \vee p)}}}, \text{ ley de De Morgan.}$$

$$\iff (\overline{p} \downarrow q) \downarrow (\overline{q} \downarrow p), \text{ equivalencia lógica NOR.}$$

$$\iff ((\overline{p \vee p}) \downarrow q) \downarrow ((\overline{q \vee q}) \downarrow p), \text{ ley de idempotencia.}$$

$$\iff ((p \downarrow p) \downarrow q) \downarrow ((q \downarrow q) \downarrow p), \text{ equivalencia lógica NOR.}$$

Ejemplo 11:

Representar la proposición $(p \wedge q) \rightarrow (q \vee r)$ sólo con el conectivo NOR (\downarrow)

Para resolver este ejercicio más fácilmente, primero se simplificará la proposición compuesta y luego sobre dicha simplificación se buscará la representación utilizando únicamente el conectivo NOR, para esto se tiene:

$$(p \wedge q) \rightarrow (q \vee r) \iff \overline{\overline{(p \wedge q)}} \vee (q \vee r), \text{ equivalencia lógica de la implicación.}$$

$$\iff (\bar{p} \vee \bar{q}) \vee (q \vee r), \text{ ley de De Morgan.}$$

$$\iff \bar{p} \vee ((\bar{q} \vee q) \vee r), \text{ ley asociativa.}$$

$$\iff \bar{p} \vee (V \vee r), \text{ ley inversa.}$$

$$\iff \bar{p} \vee V, \text{ ley de dominación.}$$

$$\iff V, \text{ ley de dominación.}$$

Ahora se transformará el resultado de la simplificación utilizando únicamente el conector NOR

$$V \iff (\bar{p} \vee p), \text{ ley inversa.}$$

$$\iff (\bar{p} \vee p) \wedge (\bar{p} \vee p), \text{ ley de idempotencia.}$$

$$\iff \overline{(\bar{p} \vee p) \wedge (\bar{p} \vee p)}, \text{ ley de doble negación.}$$

$$\iff \overline{(\bar{p} \vee p) \vee (\bar{p} \vee p)}, \text{ ley de De Morgan.}$$

$$\iff (\bar{p} \downarrow p) \downarrow (\bar{p} \downarrow p), \text{ equivalencia lógica NOR.}$$

$$\iff ((\bar{p} \vee p) \downarrow p) \downarrow ((\bar{p} \vee p) \downarrow p), \text{ ley de idempotencia.}$$

$$\iff (((\bar{p} \downarrow p) \downarrow p) \downarrow ((\bar{p} \downarrow p) \downarrow p)), \text{ equivalencia lógica NOR.}$$

Ejemplo 12:

Demostrar utilizando las leyes de la lógica proposicional que las proposiciones compuestas $(\bar{p} \wedge \bar{q}) \vee (p \leftrightarrow q)$ y $((p \downarrow p) \downarrow q) \downarrow ((p \downarrow p) \downarrow q)$ son lógicamente equivalentes.

Los pasos que justifican la demostración son los siguientes:

$$\overline{(\bar{p} \wedge \bar{q})} \vee (p \leftrightarrow q)$$

$$\iff (\bar{p} \vee q) \vee (p \leftrightarrow q), \text{ ley de De Morgan.}$$

$$\iff (\bar{p} \vee q) \vee ((p \rightarrow q) \wedge (q \rightarrow p)), \text{ equivalencia lógica doble implicación.}$$

$$\iff (\bar{p} \vee q) \vee ((\bar{p} \vee q) \wedge (\bar{q} \vee p)), \text{ equivalencia lógica de la implicación.}$$

$$\iff \bar{p} \vee q, \text{ ley de absorción.}$$

$$\iff (\bar{p} \vee q) \wedge (\bar{p} \vee q), \text{ ley de la idempotencia.}$$

$$\iff \overline{(\bar{p} \vee q) \wedge (\bar{p} \vee q)}, \text{ ley de la doble negación.}$$

$$\iff \overline{(\bar{p} \vee q) \vee (\bar{p} \vee q)}, \text{ ley de De Morgan.}$$

$$\iff (\bar{p} \downarrow q) \downarrow (\bar{p} \downarrow q), \text{ equivalencia lógica del NOR.}$$

$$\iff ((\overline{p \vee p}) \downarrow q) \downarrow ((\overline{p \vee p}) \downarrow q), \quad \text{ley de idempotencia.}$$

$$\iff ((p \downarrow p) \downarrow q) \downarrow ((p \downarrow p) \downarrow q), \quad \text{equivalencia lógica del NOR.}$$

1.1.9. Conectivo NAND (\uparrow)

El conectivo NAND es también un conectivo completo, en el sentido que, cualquier fórmula del cálculo proposicional puede ser escrita utilizando únicamente éste conectivo.

Mnemotécnicamente:

$$NAND \approx \text{Not AND} \approx \text{NO y} \approx \sim (p \wedge q) \approx \overline{(p \wedge q)} \approx p \wedge q \approx p \uparrow q$$

Ejemplo 13:

Demostrar que el conectivo NAND es un conectivo completo.

De forma similar como se demostró que el conectivo NOR es un conectivo completo, también se demuestra que el conectivo NAND es un conectivo completo, de esta forma se debe representar los conectivos \sim , \vee , \wedge , \rightarrow y \leftrightarrow , utilizando únicamente el conectivo NAND.

Representación de la negación utilizando únicamente el NAND:

$$\bar{p} \iff \overline{p \wedge p}, \quad \text{ley de idempotencia}$$

$$\iff p \uparrow p, \quad \text{equivalencia lógica de la NAND.}$$

Representación de la disyunción utilizando únicamente el NAND:

$$p \vee q \iff \overline{\overline{p \vee q}}, \quad \text{ley de la doble negación.}$$

$$\iff \overline{\bar{p} \wedge \bar{q}}, \quad \text{ley de De Morgan.}$$

$$\iff \overline{(p \wedge p) \wedge (q \wedge q)}, \quad \text{ley de idempotencia.}$$

$$\iff (p \uparrow p) \uparrow (q \uparrow q), \quad \text{equivalencia lógica NAND.}$$

Representación de la conjunción utilizando únicamente el NAND:

$$p \wedge q \iff (p \wedge q) \vee (p \wedge q), \quad \text{ley de idempotencia.}$$

$$\iff \overline{\overline{(p \wedge q) \vee (p \wedge q)}}, \quad \text{ley de doble negación.}$$

$$\iff \overline{(p \wedge q) \wedge (p \wedge q)}, \quad \text{ley de De Morgan.}$$

$$\iff (p \uparrow q) \uparrow (p \uparrow q), \text{ equivalencia lógica NAND.}$$

Representación de la implicación utilizando únicamente el NAND:

$$p \rightarrow q \iff \bar{p} \vee q, \text{ equivalencia lógica de la implicación.}$$

$$\iff \overline{(\bar{p} \vee q)}, \text{ ley de la doble negación.}$$

$$\iff \overline{(p \wedge \bar{q})}, \text{ ley de De Morgan.}$$

$$\iff \overline{(p \wedge \overline{(q \wedge q)})}, \text{ ley de idempotencia.}$$

$$\iff (p \uparrow (q \uparrow q)), \text{ equivalencia lógica NAND.}$$

Representación de la doble implicación utilizando únicamente el NAND:

$$p \leftrightarrow q \iff (p \rightarrow q) \wedge (q \rightarrow p), \text{ equivalencia lógica de la doble implicación.}$$

$$\iff (\bar{p} \vee q) \wedge (\bar{q} \vee p), \text{ equivalencia lógica de la implicación.}$$

$$\iff ((\bar{p} \vee q) \wedge \bar{q}) \vee ((\bar{p} \vee q) \wedge p), \text{ ley distributiva.}$$

$$\iff ((\bar{p} \wedge \bar{q}) \vee (q \wedge \bar{q})) \vee ((\bar{p} \wedge p) \vee (q \wedge p)), \text{ ley distributiva.}$$

$$\iff ((\bar{p} \wedge \bar{q}) \vee F) \vee (F \vee (q \wedge p)), \text{ ley inversa.}$$

$$\iff ((\bar{p} \wedge \bar{q}) \vee ((q \wedge p))), \text{ ley de identidad.}$$

$$\iff \overline{(\bar{p} \wedge \bar{q}) \vee (q \wedge p)}, \text{ ley de doble negación.}$$

$$\iff \overline{(\bar{p} \wedge \bar{q}) \wedge (q \wedge p)}, \text{ ley de De Morgan.}$$

$$\iff \overline{((\bar{p} \wedge \bar{p}) \wedge (\bar{q} \wedge \bar{q})) \wedge (q \wedge p)}, \text{ ley de idempotencia.}$$

$$\iff ((p \uparrow p) \uparrow (q \uparrow q)) \uparrow (q \uparrow p), \text{ equivalencia lógica NAND.}$$

Ejemplo 14:

Representar la proposición $\overline{(p \vee q) \otimes \bar{q}}$ utilizando sólo el conectivo NAND (\uparrow), y donde se utilice la mínima cantidad de estos.

Para utilizar la mínima cantidad de conectivos NAND, es necesario simplificar primero la proposición compuesta, para luego, sobre la simplificación buscar el equivalente utilizando únicamente el conectivo NAND. De esta forma se tiene:

$$\overline{(p \vee q) \otimes \bar{q}} \iff \overline{((\bar{p} \vee \bar{q}) \wedge \bar{q}) \vee ((p \vee q) \wedge \bar{q})}, \text{ equivalencia lógica del X-OR}$$

$$\begin{aligned}
&\iff \overline{((p \vee q) \wedge \bar{q}) \vee ((p \vee q) \wedge q)}, \quad \text{ley de doble negación} \\
&\iff \overline{((p \vee q) \wedge \bar{q})} \wedge \overline{((p \vee q) \wedge q)}, \quad \text{ley de De Morgan} \\
&\iff \overline{((p \vee q) \wedge \bar{q})} \wedge \overline{(q)}, \quad \text{ley de absorción} \\
&\iff (\overline{(p \vee q)} \vee \bar{\bar{q}}) \wedge \overline{(q)}, \quad \text{ley de De Morgan} \\
&\iff ((p \vee q) \vee q) \wedge \bar{q}, \quad \text{ley de doble negación} \\
&\iff (p \vee (q \vee q)) \wedge \bar{q}, \quad \text{ley asociativa} \\
&\iff (p \vee q) \wedge \bar{q}, \quad \text{ley de idempotencia} \\
&\iff (q \wedge \bar{q}) \vee (p \wedge \bar{q}), \quad \text{ley distributiva} \\
&\iff F \vee (p \wedge \bar{q}), \quad \text{ley inversa} \\
&\iff p \wedge \bar{q}, \quad \text{ley de identidad} \\
&\iff (p \wedge \bar{q}) \vee (p \wedge \bar{q}), \quad \text{ley de idempotencia} \\
&\iff \overline{\overline{(p \wedge \bar{q})} \vee \overline{(p \wedge \bar{q})}}, \quad \text{ley de doble negación} \\
&\iff \overline{\overline{(p \wedge \bar{q})} \wedge \overline{(p \wedge \bar{q})}}, \quad \text{ley de De Morgan} \\
&\iff (p \uparrow \bar{q}) \uparrow (p \uparrow \bar{q}), \quad \text{equivalencia lógica del NAND} \\
&\iff (p \uparrow \overline{(q \wedge q)}) \uparrow (p \uparrow \overline{(q \wedge q)}), \quad \text{ley de idempotencia} \\
&\iff (p \uparrow (q \uparrow q)) \uparrow (p \uparrow (q \uparrow q)), \quad \text{equivalencia lógica del NAND}
\end{aligned}$$

1.1.10. Ejercicios

- Determinar cuáles de las siguientes asignaciones de verdad a las proposiciones primitivas p , q , r y s hacen que la proposición compuesta $(p \wedge (q \vee r)) \rightarrow (r \wedge s)$ tome el valor falso.

- $p=V, q=V, r=F, s=V$
- $p=V, q=V, r=F, s=F$
- $p=F, q=F, r=V, s=F$
- $p=V, q=F, r=V, s=V$
- $p=V, q=F, r=V, s=F$
- $p=F, q=F, r=F, s=F$
- $p=V, q=V, r=V, s=V$

- Identificar cuáles de las siguientes proposiciones compuestas son tautologías.

- $p \rightarrow ((\neg(r \wedge s) \vee (t \leftrightarrow (v \rightarrow (\neg w \vee q)))) \vee (r \vee w)) \rightarrow p$

- b) $(p \rightarrow q) \rightarrow p$
- c) $(p \rightarrow q) \leftrightarrow (\neg p \leftrightarrow \neg q)$
- d) $\neg(\neg(\neg(\neg p \vee p) \vee p) \vee p) \vee p$
- e) $(p \rightarrow (q \vee r)) \rightarrow ((p \rightarrow q) \vee (p \rightarrow r))$

3. Simplificar las siguientes proposiciones compuestas utilizando las leyes de la lógica proposicional. En cada paso de la simplificación registrar la regla que se utilizó.

a) $p \rightarrow (p \rightarrow (p \rightarrow (\bar{p} \rightarrow (r \vee s))))$

Respuesta: V

b) $(\bar{p} \vee q) \wedge [(p \wedge \bar{q}) \vee (\bar{p} \wedge q)] \wedge (p \vee \bar{q})$

Respuesta: F

c) $(p \wedge \bar{q}) \vee (\bar{p} \wedge q) \vee [(\bar{p} \vee q) \wedge (p \vee \bar{q})]$

Respuesta: V

d) $(p \wedge q \wedge r) \vee (p \wedge t \wedge \bar{q}) \vee (p \wedge t \wedge r)$

Respuesta: $p \wedge ((q \vee t) \wedge (r \vee (t \wedge \bar{q})))$

e) $(p \wedge q \wedge r) \vee (p \wedge t \wedge \bar{q}) \vee (p \wedge \bar{t} \wedge r)$

f) $(x \otimes y) \otimes x$

Respuesta: y

g) $\overline{(x \otimes y)} \rightarrow \bar{y}$

Respuesta: $\overline{x \wedge y}$

h) $(p \downarrow q) \otimes (p \uparrow q)$

i) $(p \downarrow q) \downarrow (p \otimes q)$

j) $(p \downarrow q) \uparrow (p \otimes q)$

k) $(p \downarrow q) \leftrightarrow (p \uparrow q)$

l) $((p \uparrow q) \uparrow p) \uparrow (q \uparrow q)$

m) $((p \uparrow q) \downarrow p) \uparrow (q \downarrow q)$

n) $((p \uparrow q) \uparrow p) \uparrow q$

\tilde{n}) $((p \downarrow q) \uparrow p) \downarrow q$

o) $((p \downarrow q) \downarrow p) \downarrow (q \downarrow q)$

p) $((p \uparrow q) \downarrow p) \downarrow (q \uparrow q)$

q) $((p \downarrow q) \downarrow p) \downarrow q$

r) $((p \uparrow q) \uparrow p) \downarrow q$

4. Representar las siguientes proposiciones compuestas utilizando únicamente el conector NOR (\downarrow)

a) $(p \vee \bar{q}) \wedge (\bar{p} \vee q)$

b) $(\bar{p} \wedge \bar{q}) \vee (p \leftrightarrow q)$

Respuesta: $p \leftrightarrow q \Leftrightarrow ((p \downarrow p) \downarrow q) \downarrow ((q \downarrow q) \downarrow p)$

c) $(p \wedge \bar{q}) \vee (p \leftrightarrow q)$

Respuesta: $q \rightarrow p \Leftrightarrow ((q \downarrow q) \downarrow p) \downarrow ((q \downarrow q) \downarrow p)$

d) $(\bar{p} \wedge \bar{q}) \wedge (p \leftrightarrow q)$

e) $(p \wedge \bar{q}) \wedge (p \leftrightarrow q)$

f) $(p \wedge q) \rightarrow (q \vee r)$

Respuesta: $((p \downarrow p) \downarrow p) \downarrow ((p \downarrow p) \downarrow p)$

g) $\overline{p \Leftrightarrow q}$

h) $p \otimes q$

i) $(\overline{p \otimes q}) \vee (\bar{p} \leftrightarrow q)$

Respuesta: $(p \downarrow (p \downarrow p)) \downarrow (p \downarrow (p \downarrow p))$

5. Representar las siguientes proposiciones compuestas utilizando únicamente el conector NAND (\uparrow)

a) $(p \wedge \bar{q}) \vee (\bar{p} \vee q)$

Respuesta: $(p \uparrow p) \uparrow p$

b) $(\bar{p} \wedge \bar{q}) \vee (p \leftrightarrow q)$

c) $(p \wedge \bar{q}) \vee (p \leftrightarrow q)$

Respuesta: $(p \uparrow (q \uparrow q)) \uparrow ((p \uparrow p) \uparrow (q \uparrow q)) \uparrow (p \uparrow q)$

d) $(\bar{p} \wedge \bar{q}) \wedge (p \leftrightarrow q)$

e) $(p \wedge \bar{q}) \wedge (p \leftrightarrow q)$

f) $\overline{p \Leftrightarrow q}$

g) $(\overline{p \wedge \bar{q}}) \vee (p \otimes q)$

Respuesta: $p \uparrow (p \uparrow p)$

h) $p \otimes q$

Respuesta: $(p \uparrow (q \uparrow q)) \uparrow ((p \uparrow p) \uparrow q)$

i) $(p \uparrow q) \wedge (p \otimes q)$

j) $(p \uparrow q) \vee (p \otimes q)$

1.2. Reglas de inferencia

Las reglas de inferencia son utilizadas en la lógica proposicional, para demostrar que una conclusión se sigue lógicamente de un conjunto de hipótesis, al utilizar una serie de pasos que involucran reglas de inferencia y/o leyes de la lógica proposicional.

1.2.1. Tabla de reglas de inferencia

Regla de inferencia	Tautología	Nombre
$\frac{p}{\therefore p \vee q}$	$p \rightarrow (p \vee q)$	Adición
$\frac{p \wedge q}{\therefore p}$	$(p \wedge q) \rightarrow p$	Simplificación
$\frac{p}{q} \quad \frac{q}{\therefore p \wedge q}$	$[(p) \wedge (q)] \rightarrow (p \wedge q)$	Conjunción
$\frac{p}{p \rightarrow q} \quad \frac{p \rightarrow q}{\therefore q}$	$[p \wedge (p \rightarrow q)] \rightarrow q$	Modus Ponens
$\frac{\neg q}{p \rightarrow q} \quad \frac{p \rightarrow q}{\therefore \neg p}$	$[\neg q \wedge (p \rightarrow q)] \rightarrow \neg p$	Modus Tollens
$\frac{p \rightarrow q}{q \rightarrow r} \quad \frac{q \rightarrow r}{\therefore p \rightarrow r}$	$[(p \rightarrow q) \wedge (q \rightarrow r)] \rightarrow (p \rightarrow r)$	Silogismo Hipotético
$\frac{p \vee q}{\neg p} \quad \frac{\neg p}{\therefore q}$	$[(p \vee q) \wedge \neg p] \rightarrow q$	Silogismo Disyuntivo
$\frac{p \vee q}{\neg p \vee r} \quad \frac{\neg p \vee r}{\therefore q \vee r}$	$[(p \vee q) \wedge (\neg p \vee r)] \rightarrow (q \vee r)$	Resolución

Ejemplo 15:

El siguiente razonamiento es un ejemplo de la regla de inferencia de adición:

“Antonio es una persona joven. Por lo tanto, Antonio es una persona joven o Antonio es una persona saludable”.

A partir del enunciado del razonamiento se obtienen las proposiciones:

p : Antonio es una persona joven

q : Antonio es una persona saludable

El razonamiento puede ser reescrito ahora como la regla de inferencia de adición:

$$\frac{p}{\therefore p \vee q}$$

Ejemplo 16:

El siguiente razonamiento es un ejemplo de la regla de inferencia de simplificación:

“Antonio es una persona joven y Antonio es una persona saludable. Por lo tanto, Antonio es una persona joven”.

A partir del enunciado del razonamiento se obtienen las proposiciones:

p : Antonio es una persona joven

q : Antonio es una persona saludable

El razonamiento puede ser reescrito ahora como la regla de inferencia de simplificación:

$$\frac{p \wedge q}{\therefore p}$$

Ejemplo 17:

El siguiente razonamiento es un ejemplo de la regla de inferencia de conjunción:

“Rosa es elegida presidenta de la junta de acción comunal. Elena ingresa a la junta de acción comunal. Por lo tanto, Rosa es elegida presidenta de la junta de acción comunal y Elena ingresa a la junta de acción comunal”.

A partir del enunciado del razonamiento se obtienen las proposiciones:

p : Rosa es elegida presidenta de la junta de acción comunal

q : Elena ingresa a la junta de acción comunal

El razonamiento puede ser reescrito ahora como la regla de inferencia de conjunción:

$$\frac{p \quad q}{\therefore p \wedge q}$$

Ejemplo 18:

El siguiente razonamiento es un ejemplo de la regla de inferencia de Modus Ponens:

“Si Lina gana 100 millones de pesos en la lotería, entonces, José renunciará a su trabajo. Lina ganó 100 millones de pesos en la lotería. Por lo tanto, José renunciará a su trabajo”.

A partir del enunciado del razonamiento se obtienen las proposiciones:

p : Lina gana 100 millones de pesos en la lotería
 q : José renuncia a su trabajo

El razonamiento puede ser reescrito ahora como la regla de inferencia de Modus Ponens:

$$\frac{p \rightarrow q \quad p}{\therefore q}$$

Ejemplo 19:

El siguiente razonamiento es un ejemplo de la regla de inferencia de Modus Tollens:

“Si Lina gana 100 millones de pesos en la lotería, entonces, José renunciará a su trabajo. Se sabe que José no renunció a su trabajo. Por lo tanto, Lina no ganó 100 millones de pesos en la lotería”.

A partir del enunciado del razonamiento se obtienen las proposiciones:

p : Lina gana 100 millones de pesos en la lotería
 q : José renuncia a su trabajo

El razonamiento puede ser reescrito ahora como la regla de inferencia de Modus Tollens:

$$\frac{p \rightarrow q \quad \neg q}{\therefore \neg p}$$

Ejemplo 20:

El siguiente razonamiento es un ejemplo de la regla de inferencia de silogismo hipotético:

“Si hoy es un día lluvioso, entonces no debemos tener un asado hoy. Si no debemos tener un asado hoy, entonces debemos tener un asado mañana, Por lo tanto, si hoy es un día lluvioso, entonces debemos tener un asado mañana”.

A partir del enunciado del razonamiento se obtienen las proposiciones:

p : Hoy es un día lluvioso

q : No debemos tener un asado hoy

r : Debemos tener el asado mañana

El razonamiento puede ser reescrito ahora como la regla de inferencia de silogismo hipotético:

$$\begin{array}{l} p \rightarrow q \\ q \rightarrow r \\ \hline \therefore p \rightarrow r \end{array}$$

Ejemplo 21:

El siguiente razonamiento es un ejemplo de la regla de inferencia de silogismo disyuntivo:

“La billetera de Carlos está en su bolsillo o la billetera de Carlos está en la mesa. Se sabe que la billetera de Carlos no está en su bolsillo. Por lo tanto, la billetera de Carlos está en la mesa”.

A partir del enunciado del razonamiento se obtienen las proposiciones:

p : La billetera de Carlos está en su bolsillo

q : La billetera de Carlos está en la mesa

El razonamiento puede ser reescrito ahora como la regla de inferencia de silogismo disyuntivo:

$$\begin{array}{l} p \vee q \\ \neg p \\ \hline \therefore q \end{array}$$

1.2.2. Utilización de las reglas de inferencia para demostrar la validez de razonamientos

Ejemplo 22:

Determinar si el siguiente razonamiento es válido:

$$\begin{array}{l}
 \bar{g} \\
 \bar{g} \rightarrow e \\
 e \rightarrow k \\
 k \rightarrow \bar{l} \\
 \bar{l} \rightarrow m \\
 m \rightarrow b \\
 \hline
 \therefore b
 \end{array}$$

En la solución de todos los ejemplos de esta sección, se numeran cada una de las hipótesis, para después poder referenciar a cada una de las éstas en las razones que justifican cada uno de los resultados en los pasos. Por ejemplo, para referenciar a la hipótesis 4 se utiliza H_4 . De forma similar, para referenciar el resultado del paso i del análisis, se utilizará P_i .

La numeración de las hipótesis es la siguiente:

$$\begin{array}{l}
 1. \quad \bar{g} \\
 2. \quad \bar{g} \rightarrow e \\
 3. \quad e \rightarrow k \\
 4. \quad k \rightarrow \bar{l} \\
 5. \quad \bar{l} \rightarrow m \\
 6. \quad m \rightarrow b \\
 \hline
 \therefore b
 \end{array}$$

Los pasos, resultados y razones que se necesitan para demostrar la validez del razonamiento son los siguientes:

Pasos	Resultados	Razones
1.	e	Regla Modus Ponens entre H_1 e H_2
2.	k	Regla Modus Ponens entre P_1 e H_3
3.	\bar{l}	Regla Modus Ponens entre P_2 e H_4
4.	m	Regla Modus Ponens entre P_3 e H_5
5.	b	Regla Modus Ponens entre P_4 e H_6

Como la conclusión se obtiene a partir de las hipótesis y de la utilización de las reglas

de inferencia, entonces se concluye que el razonamiento es válido.

Ejemplo 23:

Mostrar o refutar que las hipótesis $p \rightarrow (q \rightarrow r)$, $\bar{q} \rightarrow \bar{p}$ y p implican la conclusión r .

El razonamiento se puede reescribir como:

1. $p \rightarrow (q \rightarrow r)$
 2. $\bar{q} \rightarrow \bar{p}$
 3. p
-
- $\therefore r$

Los pasos y razones necesarios para obtener la conclusión a partir de las hipótesis, son:

Pasos	Resultados	Razones
1.	q	Regla Modus Tollens entre H_2 e H_3
2.	$q \rightarrow r$	Regla Modus Ponens entre H_1 e H_3
3.	r	Regla Modus Ponens entre P_1 y P_2

El razonamiento es válido porque la conclusión se obtiene a partir las hipótesis y del uso de algunas de las reglas de inferencia.

Ejemplo 24:

En el siguiente razonamiento ya se han enumerado las hipótesis, determinar su validez.

1. $p \rightarrow r$
 2. $p \rightarrow q$
-
- $\therefore p \rightarrow (r \wedge q)$

La justificación de la validez del razonamiento se apoya en los siguientes pasos, resultados y razones:

Pasos	Resultados	Razones
1.	$\bar{p} \vee r$	Equivalencia lógica de la implicación en H_1
2.	$\bar{p} \vee q$	Equivalencia lógica de la implicación en H_2
3.	$(\bar{p} \vee r) \wedge (\bar{p} \vee q)$	Regla de Conjunción entre P_1 y P_2
4.	$\bar{p} \vee (r \wedge q)$	Ley distributiva en P_3
5.	$p \rightarrow (r \wedge q)$	Equivalencia lógica de la implicación en P_4

Utilizando las leyes de la lógica, las reglas de inferencia y las hipótesis originales, se obtiene la conclusión del razonamiento, por lo tanto el razonamiento es válido.

Ejemplo 25:

Determinar si el siguiente razonamiento es válido. En el razonamiento las hipótesis ya están numeradas.

1. $p \rightarrow q$
2. $q \rightarrow s$
3. $r \rightarrow \bar{s}$
4. $\frac{\bar{p} \otimes r}{\therefore \bar{r}}$

Los pasos, resultados y razones necesarios para deducir la conclusión a partir de las hipótesis, el uso de las leyes de la lógica proposicional y del uso de las reglas de inferencia, son:

Pasos	Resultados	Razones
1.	$(p \wedge r) \vee (\bar{p} \wedge \bar{r})$	Equivalencia lógica del X-OR en H_4
2.	$p \rightarrow s$	Regla de Silogismo hipotético entre H_1 e H_2
3.	$\bar{r} \vee \bar{s}$	Equivalencia lógica de la implicación en H_3
4.	$\bar{s} \vee \bar{r}$	Ley conmutativa en P_3
5.	$s \rightarrow \bar{r}$	Equivalencia lógica de la implicación en P_4
6.	$p \rightarrow \bar{r}$	Regla de Silogismo Hipotético entre P_2 y P_5
7.	$\bar{p} \vee \bar{r}$	Equivalencia lógica de la implicación en P_6
8.	$\overline{(p \wedge r)}$	Ley de De Morgan en P_7
9.	$\bar{p} \wedge \bar{r}$	Regla de Silogismo Disyuntivo entre P_1 y P_8
10.	\bar{r}	Regla de simplificación en P_9

Se obtiene la conclusión, por lo tanto el razonamiento es válido.

Ejemplo 26:

Determinar si el siguiente razonamiento es válido:

- $$\begin{array}{l}
 (p \wedge q) \vee r \\
 r \rightarrow s \\
 (p \vee s) \rightarrow \bar{t} \\
 (q \vee s) \rightarrow u \\
 \hline
 \therefore \bar{u} \rightarrow \bar{t}
 \end{array}$$

La numeración de las hipótesis en el razonamiento es el siguiente:

1. $(p \wedge q) \vee r$
 2. $r \rightarrow s$
 3. $(p \vee s) \rightarrow \bar{t}$
 4. $(q \vee s) \rightarrow u$
-
- $\therefore \bar{u} \rightarrow \bar{t}$

La validez del razonamiento se obtiene con los siguientes pasos, resultados y razones:

Pasos	Resultados	Razones
1.	$(p \vee r) \wedge (q \vee r)$	Ley distributiva en H_1
2.	$p \vee r$	Regla de Simplificación del P_1
3.	$\bar{p} \rightarrow r$	Equivalencia lógica de la implicación del P_2
4.	$\bar{p} \rightarrow s$	Regla de Silogismo Hipotético entre P_3 e H_2
5.	$p \vee s$	Equivalencia lógica de la implicación del P_4
6.	\bar{t}	Regla Modus Ponens entre P_5 e H_3
7.	$q \vee r$	Regla de Simplificación del P_1
8.	$\bar{q} \rightarrow r$	Equivalencia lógica de la implicación del P_7
9.	$\bar{q} \rightarrow s$	Regla de Silogismo Hipotético entre P_8 e H_2
10.	$q \vee s$	Equivalencia lógica de la implicación del P_9
11.	u	Regla Modus Ponens entre P_{10} e H_4
12.	$u \vee \bar{t}$	Regla de Adición entre P_{11} y P_6
13.	$\bar{u} \rightarrow \bar{t}$	Equivalencia lógica de la implicación del P_{12}

Se obtiene la conclusión a partir de las hipótesis, de la aplicación de las leyes de la lógica proposicional y de reglas de inferencia. De esta forma la validez del razonamiento queda demostrada, pero, el resultado del paso número 12 a pesar de que es correcto no se ve muy “natural”. Por este motivo la validez de este razonamiento se demostrará de nuevo en el capítulo 3 de Técnicas de Demostración en la subsección 3.2.2 de Técnica de Demostración por Contradicción, donde se obtiene una demostración más natural y aceptable para este razonamiento.

Ejemplo 27:

Mostrar o refutar que las hipótesis $q \vee p$, $\overline{p \vee \bar{r}}$, $r \rightarrow s$ y $(q \wedge s) \rightarrow (t \wedge s)$ implican la conclusión t .

El razonamiento se puede reescribir numerando las hipótesis de la siguiente forma:

1. $q \vee p$
2. $\frac{p \vee \bar{r}}{p \vee \bar{r}}$
3. $r \rightarrow s$
4. $\frac{(q \wedge s) \rightarrow (t \wedge s)}{\therefore t}$

Una forma correcta para establecer la validez del razonamiento es la siguiente:

Pasos	Resultados	Razones
1.	$\bar{p} \wedge r$	Ley de De Morgan en H_2
2.	\bar{p}	Regla de Simplificación del P_1
3.	r	Regla de Simplificación del P_1
4.	s	Regla Modus Ponens entre P_3 e H_3
5.	q	Regla de Silogismo disyuntivo entre P_2 e H_1
6.	$q \wedge s$	Regla de Conjunción entre P_5 y P_4
7.	$t \wedge s$	Regla Modus Ponens entre P_6 e H_4
8.	t	Regla de Simplificación del P_7

Se obtiene la conclusión del razonamiento a partir del conjunto de hipótesis, y del adecuado uso tanto, de las leyes de la lógica proposicional, como de las reglas de inferencia. Por lo tanto el razonamiento es correcto.

Ejemplo 28:

Determinar si el siguiente razonamiento es válido:

1. $u \rightarrow r$
 2. $(r \wedge s) \rightarrow (p \vee t)$
 3. $q \rightarrow (u \wedge s)$
 4. \bar{t}
-
- $\therefore q \rightarrow p$

Los pasos y razones que se necesitan para demostrar la validez del razonamiento son los siguientes:

Pasos	Resultados	Razones
1.	$\overline{(r \wedge s)} \vee (p \vee t)$	Equivalencia lógica de la implicación en H_2
2.	$\overline{((r \wedge s) \vee p)} \vee t$	Ley asociativa en P_1
3.	$\overline{((r \wedge s) \vee p)}$	Regla de Silogismo Disyuntivo entre P_2 e H_4
4.	$(\bar{r} \vee \bar{s}) \vee p$	Ley de De Morgan en P_3
5.	$\bar{r} \vee (\bar{s} \vee p)$	Ley asociativa en P_4
6.	$r \rightarrow (\bar{s} \vee p)$	Equivalencia lógica de la implicación en P_5
7.	$u \rightarrow (\bar{s} \vee p)$	Regla de Silogismo Hipotético entre H_1 y P_6
8.	$\bar{u} \vee (\bar{s} \vee p)$	Equivalencia lógica de la implicación en P_7
9.	$\overline{(\bar{u} \vee \bar{s})} \vee p$	Ley asociativa en P_8
10.	$\overline{(\bar{u} \vee \bar{s})} \rightarrow p$	Equivalencia lógica de la implicación en P_9
11.	$(u \wedge s) \rightarrow p$	Ley de De Morgan en P_{10}
12.	$q \rightarrow p$	Regla de Silogismo Hipotético entre H_3 y P_{11}

Se obtiene la conclusión a partir de las hipótesis, de la aplicación de las leyes de la lógica proposicional y de las reglas de inferencia, de esta forma queda demostrada la validez del razonamiento. Este ejemplo, también se demostrará de nuevo en el capítulo 3 de Técnicas de Demostración, utilizando una método diferente.

1.2.3. Ejercicios

1. Escribir cada uno de los siguientes argumentos en forma simbólica, después determine por reglas de inferencia si cada uno de éstos es válido:
 - a) Si Carlos va a la carrera de autos, entonces Elena se enojará. Si Rafael juega cartas toda la noche, entonces Carmen se enojará. Si Elena o Carmen se enojan, le avisarán a Verónica (su abogado). Verónica no ha tenido noticias de estas dos clientes. En consecuencia, ni Carlos fue a la carrera ni Rafael jugó cartas toda la noche.
 - b) Si Rosa María obtiene el puesto de supervisor y trabaja mucho, entonces obtendrá un aumento. Si obtiene el aumento, entonces comprará un auto nuevo. Ella no ha adquirido un auto nuevo. Por lo tanto, Rosa María no ha obtenido el puesto de supervisor o no ha trabajado mucho.
 - c) Si la banda no pudiera tocar rock o las bebidas no llegasen a tiempo, entonces la fiesta de Año Nuevo tendría que cancelarse y Alicia se enojaría. Si la fiesta se cancelara, habría que devolver el dinero. No se devolvió el dinero. Por lo tanto, la banda pudo tocar rock.
 - d) Si Tomás tiene 17 años, entonces es de la misma edad de Juana. Si José no tiene la misma edad de Tomás, entonces José tiene distinta edad que Juana. Tomás tiene 17 años y José tiene la misma edad que Juana. Por lo tanto, José tiene la misma edad que Tomás y Tomás tiene la misma edad que Juana.

- e) Si es verdad que si llueve entonces los estudiantes se acuestan, entonces los estudiantes no estudian. Si los estudiantes aprueban el examen entonces o los estudiantes estudian o el examen es trivial. Si el examen es trivial, entonces los estudiantes son flojos. Es un hecho que los estudiantes aprueban el examen y no son flojos. Por lo tanto, llueve y los estudiantes no se acuestan.
- f) Si el contrato es legal y Pérez entró en el contrato, entonces García ganará el pleito. O García no ganará el pleito o Pérez será responsable. Pérez no será responsable. Por lo tanto, o el contrato no es legal o Pérez no entró en el contrato.
2. Determinar si cada uno de los siguientes razonamientos es válido. Para cada uno de los pasos del análisis, indicar, que regla de inferencia o ley de la lógica proposicional se utilizó y sobre que hipótesis o resultados intermedios se aplicó.

$$\begin{array}{l}
 a) \quad (\bar{q} \rightarrow \bar{p}) \\
 \quad (q \rightarrow \bar{r}) \\
 \quad \underline{(\bar{p} \wedge r)} \\
 \quad \therefore \bar{r}
 \end{array}$$

$$\begin{array}{l}
 b) \quad t \rightarrow (p \wedge s) \\
 \quad q \rightarrow \bar{p} \\
 \quad r \rightarrow \bar{s} \\
 \quad \underline{r \vee q} \\
 \quad \therefore \bar{t}
 \end{array}$$

$$\begin{array}{l}
 c) \quad \overline{r \wedge \bar{t}} \\
 \quad \bar{q} \vee r \\
 \quad \bar{q} \rightarrow \bar{p} \\
 \quad p \\
 \quad \underline{\bar{q} \vee u \vee \bar{t}} \\
 \quad \therefore u
 \end{array}$$

$$\begin{array}{l}
 d) \quad p \\
 \quad p \rightarrow q \\
 \quad r \rightarrow \bar{q} \\
 \quad \underline{s \vee r} \\
 \quad \therefore s \vee t
 \end{array}$$

$$\begin{array}{l}
 e) \quad (p \wedge q) \vee r \\
 \quad (r \rightarrow s) \\
 \hline
 \therefore (p \vee s)
 \end{array}$$

$$\begin{array}{l}
 f) \quad \bar{p} \leftrightarrow q \\
 \quad q \rightarrow r \\
 \quad \bar{r} \\
 \hline
 \therefore p
 \end{array}$$

$$\begin{array}{l}
 g) \quad (x = 5) \vee (x < y) \\
 \quad ((x > 3) \vee (z < 2)) \rightarrow ((z < x) \vee (y = 1)) \\
 \quad (x < y) \rightarrow (z < 2) \\
 \quad (x = 5) \rightarrow (x > 3) \\
 \quad (z < x) \rightarrow (x = 4) \\
 \quad (y = 1) \rightarrow ((x > 3) \vee (z < 2)) \\
 \hline
 \therefore (x = 4)
 \end{array}$$

$$\begin{array}{l}
 h) \quad (x > y) \vee (x < 6) \\
 \quad (x > y) \rightarrow (x > 4) \\
 \quad (x > 4) \rightarrow ((x = 5) \wedge (x < 7)) \\
 \quad (x < 6) \rightarrow ((x = 5) \wedge (x < 7)) \\
 \quad ((x < 7) \wedge (x = 5)) \rightarrow ((z > x) \vee (y < z)) \\
 \quad (x > y) \rightarrow ((y < z) \vee (z > x)) \\
 \hline
 \therefore (x < 6)
 \end{array}$$

1.3. Lógica de predicados

El cálculo de predicados aparece gracias a la limitante de la lógica proposicional para representar situaciones como:

- Todo Hombre es mortal.
- Ningún número par divide a todos los números mayores o iguales a el.
- Juan ama a todas las mujeres hermosas.
- Existen personas menores de 30 años que sufren de hipertensión arterial.

Es necesario enriquecer a la lógica proposicional con elementos tales como:

- Funciones proposicionales (o predicados). Al ser funciones es necesario definir las variables de las funciones proposicionales y los dominios de dichas variables, que son conocidos con el nombre de universo del discurso.
- Cuantificadores universales y existenciales y el alcance de estos.

Esta lógica proposicional enriquecida es lo que se conoce como cálculo de predicados o lógica de predicados.

Es normal encontrar expresiones que involucran variables tales como:

- $x > 3$
- $x = y + 3$
- $x = y + z$

las cuales son frecuentemente encontradas en declaraciones matemáticas y en programas de computador.

Cuando los valores de las variables no son especificados, estas expresiones no son ni verdaderas ni falsas.

Ejemplo 29:

La expresión $x > 3$ se puede representar como $P(x) : x > 3$, donde la variable x es el sujeto de la expresión y “Es mayor que 3” es el predicado o propiedad que el sujeto de la expresión puede tener y que es representado por $P(x)$. La expresión $P(x)$ tomará el valor de verdad de la función proposicional P evaluada en x .

Una vez que le es asignado un valor a la variable x , la expresión $P(x)$ se convierte en una proposición y tiene un valor de verdad determinado.

Ejemplo 30:

Sea el predicado $P(x) : x > 3$, donde el universo del discurso (dominio de las x 's) es el conjunto de los números naturales. ¿Cuáles son los valores de verdad de $P(2)$ y $P(6)$?

- $P(2) : 2 > 3$, como es falso que 2 sea mayor que 3 entonces $P(2)$ es Falso.
- $P(6) : 6 > 3$, como es verdadero que 6 sea mayor que 3 entonces $P(6)$ es Verdadero.

En general una expresión que involucra las n variables $x_1, x_2, x_3, \dots, x_n$ puede ser denotada por $P(x_1, x_2, x_3, \dots, x_n)$, de esta forma se deja en evidencia que un predicado puede estar definido en términos de más de una variable.

1.3.1. Cuantificador universal

La cuantificación universal de $P(x)$ es la proposición “para todo x del universo del discurso, $P(x)$ ”, la cual es representada con la notación $\forall_x P(x)$. Aquí \forall es llamado cuantificador universal.

Ejemplo 31:

Sea el predicado $V(x) : x$ es una vocal, donde el universo del discurso es el conjunto de letras del alfabeto español. ¿Cuál es la interpretación en palabras y el valor de verdad de la expresión $\forall_x V(x)$?

Para la expresión $\forall_x V(x)$ la interpretación en palabras es: “Toda letra del alfabeto español es una vocal”, cuyo valor de verdad es falso, porque las letras consonantes no son vocales.

Ejemplo 32:

Representar la siguiente expresión en el cálculo de predicados: “Todo lo que brilla es oro”.

Sean los predicados $B(x) : x$ brilla, $O(x) : x$ es oro, donde el universo del discurso es el conjunto de metales; de esta forma la expresión “Todo lo que brilla es oro” se puede representar por: $\forall_x (B(x) \rightarrow O(x))$.

Ejemplo 33:

Representar por medio del cálculo de predicados la expresión: “Todo estudiante universitario en Colombia ha estudiado trigonometría”.

Sea el predicado $P(x) : x$ ha estudiado trigonometría, donde el universo del discurso es el conjunto de los estudiantes universitarios colombianos. Entonces apoyados en el predicado anterior y en el cuantificador universal se tiene que la expresión: “Todo estudiante universitario en Colombia ha estudiado trigonometría” se representa con $\forall_x P(x)$.

La expresión anterior también se puede representar en cálculo de predicados de la siguiente forma: $\forall_x (U(x) \rightarrow T(x))$, donde el universo del discurso es el conjunto de todos los estudiantes colombianos, y se tienen los siguientes predicados:

- $U(x) : x$ es un estudiante universitario.
- $T(x) : x$ ha estudiado trigonometría.

Ejemplo 34:

Sea el predicado $Q(x) : x < 2$, donde el universo del discurso es el conjunto de los números reales. ¿Cuál es el valor de verdad de $\forall_x Q(x)$?

$Q(x)$ no es verdadera para todos los números reales x , por ejemplo $Q(4)$ es falsa, por lo tanto $\forall_x Q(x)$ es falsa.

Cuando todos los elementos del universo del discurso pueden ser listados, por ejemplo, $x_1, x_2, x_3, \dots, x_n$ se tiene que el valor de verdad de $\forall_x P(x)$ es el mismo que el que se obtiene por la conjunción $P(x_1) \wedge P(x_2) \wedge P(x_3) \wedge \dots \wedge P(x_n)$, de este modo $\forall_x P(x)$ solo es verdadero si todo $P(x_i)$ es verdadero, donde $1 \leq i \leq n$.

Ejemplo 35:

¿Cuál es el valor de verdad de $\forall_x P(x)$, donde se tiene el predicado $P(x) : x^2 < 10$ y el universo del discurso consiste del conjunto de los números entero positivos que no sobrepasen al 4?

La expresión $\forall_x P(x)$ toma el mismo valor de verdad que la conjunción de las proposiciones:

$$\begin{aligned}
 P(1) \wedge P(2) \wedge P(3) \wedge P(4) &\iff (1^2 < 10) \wedge (2^2 < 10) \wedge (3^2 < 10) \wedge (4^2 < 10) \\
 &\iff (1 < 10) \wedge (4 < 10) \wedge (9 < 10) \wedge (16 < 10) \\
 &\iff (V_o) \wedge (V_o) \wedge (V_o) \wedge (F_o) \\
 &\iff F_o
 \end{aligned}$$

Entonces se tiene que $\forall_x P(x)$ es falsa en éste universo del discurso.

1.3.2. Cuantificador existencial

La cuantificación existencial de $P(x)$ es la proposición “Existe un x del universo del discurso tal que $P(x)$ ”. La notación $\exists_x P(x)$ denota la cuantificación existencial de $P(x)$. Aquí \exists es llamado el cuantificador existencial. La expresión $\exists_x P(x)$ es también expresada como: “Hay un x tal que $P(x)$ ”, “hay al menos un x tal que $P(x)$ ” o “Para algún x , $P(x)$ ”.

Ejemplo 36:

Sea el predicado $V(x) : x$ es una vocal, donde el universo del discurso es el conjunto de letras del alfabeto español. ¿Cuál es la interpretación en palabras y el valor de verdad de la expresión $\exists_x V(x)$?

Para la expresión $\exists_x V(x)$ la interpretación en palabras es: “Algunas letras del alfabeto español son vocales”, cuyo valor de verdad es cierto.

Ejemplo 37:

Sea el predicado $Q(x) : x = x + 1$, ¿Cuál es el valor de verdad de $\exists_x Q(x)$ donde el universo del discurso es el conjunto de los números reales?

Como $Q(x)$ es falsa para todo número real x entonces $\exists_x Q(x)$ es falsa.

Cuando todos los elementos del universo del discurso pueden ser listados, por ejemplo $x_1, x_2, x_3, \dots, x_n$, se tiene que el valor de verdad de $\exists_x P(x)$ es el mismo que el que se obtiene por la disyunción $P(x_1) \vee P(x_2) \vee P(x_3) \vee \dots \vee P(x_n)$; de este modo $\exists_x P(x)$ es verdadero cuando al menos un solo $P(x_i)$ es verdadero, donde $1 \leq i \leq n$.

Ejemplo 38:

¿Cuál es el valor de verdad de $\exists_x P(x)$, donde se tiene el predicado $P(x) : x^2 > 10$ y el universo del discurso consiste del conjunto de los números entero positivos que no sobrepasen al 4?

La expresión $\exists_x P(x)$ toma el mismo valor de verdad que la disyunción de las proposiciones:

$$\begin{aligned} P(1) \vee P(2) \vee P(3) \vee P(4) &\iff (1^2 > 10) \vee (2^2 > 10) \vee (3^2 > 10) \vee (4^2 > 10) \\ &\iff (1 > 10) \vee (4 > 10) \vee (9 > 10) \vee (16 > 10) \\ &\iff (F_o) \vee (F_o) \vee (F_o) \vee (V_o) \\ &\iff V_o \end{aligned}$$

Entonces se tiene que $\exists_x P(x)$ es verdadera en éste universo del discurso.

Ejemplo 39:

Se tienen los siguientes predicados $C(x)$: x tiene un computador, $A(x, y)$: x y y son amigos. El universo del discurso para las variables de los predicados es el conjunto de estudiantes de la institución. El significado en palabras de la expresión: $\forall_x(C(x) \vee \exists_y(A(x, y) \wedge C(y)))$, es: “Todo estudiante de la institución tiene computador o es amigo de algún estudiante de la institución que tiene computador”.

Ejemplo 40:

Se tiene el siguiente predicado $S(x, y)$: $x + y = 0$, donde el universo del discurso de las variables del predicado es el conjunto de los números reales. El significado en palabras de la expresión: $\forall_x \exists_y S(x, y)$, es: “Para cada número real x existe un número real y tal que la suma de x y y es igual a cero”.

Ejemplo 41:

Sea el predicado $T(x, y)$: $x \geq y$, donde el universo del discurso es el conjunto $\{1, 2, 3, 4, 5\}$.

A continuación se presentan las interpretaciones en palabras y los valores de verdad de algunas de las formas como se puede cuantificar el predicado $T(x, y)$

- Para la expresión $\forall_x \forall_y T(x, y)$ la interpretación en palabras es: “Todo número entre 1 y 5 es mayor o igual a todo número entre 1 y 5”, cuyo valor de verdad es falso.
- Para la expresión $\forall_x \exists_y T(x, y)$ la interpretación en palabras es: “ Todo número entre 1 y 5 es mayor o igual a algún número entre 1 y 5”, cuyo valor de verdad es cierto.
- Para la expresión $\exists_x \forall_y T(x, y)$ la interpretación en palabras es: “Existe un número entre 1 y 5 que es mayor o igual a todos los números entre 1 y 5”, cuyo valor de verdad es cierto.
- Para la expresión $\exists_x \exists_y T(x, y)$ la interpretación en palabras es: “Existe un número entre 1 y 5 que es mayor o igual a algún número entre 1 y 5”, cuyo valor de verdad es cierto.
- Para la expresión $\forall_y \exists_x T(x, y)$ la interpretación en palabras es: “Todo número entre 1 y 5 es menor o igual a algún número entre 1 y 5”, cuyo valor de verdad es cierto.
- Para la expresión $\exists_x \forall_y T(y, x)$ la interpretación en palabras es: “Existe un número entre 1 y 5 que es menor o igual a todo número entre 1 y 5”, cuyo valor de verdad es cierto.

1.3.3. Variables ligadas

Cuando un cuantificador es usado sobre la variable x , o cuando le es asignado un valor a dicha variable, se dice que la ocurrencia de la variable está ligada. Una ocurrencia de una variable que no está ligada por un cuantificador o no le ha sido asignada un valor particular se dice que es libre.

Todas las variables que aparecen en una función proposicional deben estar ligadas para que dicha función proposicional sea transformada en una proposición. Esto se puede realizar usando una combinación de cuantificadores universales, cuantificadores existenciales y la asignación de valores a las variables.

1.3.4. Alcance de un cuantificador

La parte de una expresión lógica para la cual un cuantificador es aplicado es llamado el alcance del cuantificador, adicionalmente, una variable también es libre si está fuera del alcance de todos los cuantificadores en la fórmula donde aparece dicha variable.

Ejemplo 42:

Se tiene el siguiente predicado $M(x, y) : x > y$, donde el universo del discurso es el conjunto de los números naturales. En la expresión $\exists_x M(x, y)$ la variable x está ligada por el cuantificador existencial, mientras que la variable y está libre porque no está ligada por un cuantificador y no le ha sido asignado ningún valor.

Ejemplo 43:

Sea la siguiente expresión del cálculo de predicados: $\forall_x [\exists_y (P(x, y) \wedge Q(x, y)) \rightarrow R(x)]$, los diferentes alcances de los cuantificadores son presentados en la siguiente expresión:

$$\underbrace{\forall_x \underbrace{[\exists_y (P(x, y) \wedge Q(x, y)) \rightarrow R(x)]}_{\text{Alcance de } \exists_y}}_{\text{Alcance de } \forall_x}$$

en la expresión no hay ninguna variable libre, todas están ligadas.

Ejemplo 44:

Sean los predicados $M(x, y) : x \leq y$, $I(x, y) : x + y = 10$, el universo del discurso de las variables de los predicados es el conjunto de los números enteros. ¿En la siguiente expresión del cálculo de predicados $\forall_x (\exists_y M(x, y) \wedge I(x, y))$ todas las ocurrencias de la variable y están ligadas?

Según la expresión, la variable y en el predicado $I(x, y)$ no está ligada porque el alcance del cuantificador existencial es únicamente el predicado $M(x, y)$.

Ejemplo 45:

Sea el siguiente predicado $M(x, y) : x < y$, donde el universo del discurso es el conjunto de los números enteros (\mathbb{Z}). ¿Cuál es la interpretación en palabras y el valor de verdad de $\forall_x \exists_x M(x, x)$?

La interpretación en palabras de $\forall_x \exists_x M(x, x)$ es: “Para todo número entero dicho número es estrictamente menor que el mismo”, el valor de verdad de esta interpretación es falso.

El ejemplo anterior deja en evidencia un serio problema que se puede presentar en el cálculo de predicados, donde la misma variable x es cuantificada de forma diferente bajo el alcance de la misma variable para otro cuantificador, esto siempre se debe evitar.

Ejemplo 46:

Sea el predicado $N(x, y) : x \leq y$, donde el universo del discurso es el conjunto de los números enteros (\mathbb{Z}). ¿Cuál es la interpretación en palabras y el valor de verdad de $\exists_x \forall_y N(x, y)$?

La interpretación en palabras de $\exists_x \forall_y N(x, y)$ es: “Existe un número entero que es menor o igual que todos los números enteros”, cuyo valor de verdad es falso porque los números enteros no están acotados inferiormente.

Ejemplo 47:

Sea el predicado $N(x, y) : x \leq y$, donde el universo del discurso es el conjunto de los números naturales (\mathbb{N}). ¿Cuál es la interpretación en palabras y el valor de verdad de $\exists_x \forall_y N(x, y)$?

La interpretación en palabras de $\exists_x \forall_y N(x, y)$ es: “Existe un número natural que es menor o igual a todos los números naturales”, cuyo valor de verdad es cierto porque los números naturales están acotados inferiormente, dicho número es el cero.

1.3.5. Negaciones y cuantificadores

Es normal tener la negación de una expresión cuantificada, por ejemplo la negación de la siguiente expresión: “Todo estudiante universitario colombiano ha estudiado trigonometría” sería: “No es el caso que cada estudiante universitario colombiano ha estudiado trigonometría”, o “Hay algún estudiante universitario en Colombia que no ha estudiado trigonometría”.

Si se tiene el predicado $P(x) : x$ ha estudiado trigonometría, donde el universo del discurso son los estudiantes universitarios colombianos, entonces $\neg \forall_x (P(x)) = \exists_x (\neg P(x))$.

Ejemplo 48:

¿Cuál es la negación de la siguiente expresión: “Hay un político honesto”?

Sea el predicado $H(x) : x$ es honesto, donde el universo del discurso es el conjunto de los políticos, la expresión: “Hay un político honesto” puede ser representada por $\exists x H(x)$, luego la negación es $\neg \exists x H(x)$ que significa: “Ningún político es honesto”, o que puede también ser visto como: $\forall x \neg H(x)$ que significa: “Todo político es deshonesto”.

Ejemplo 49:

¿Cuál es la negación de la siguiente expresión $\forall x (x^2 \geq x)$?, donde el universo del discurso es el conjunto de los números enteros.

Primero que todo el significado en palabras de la expresión $\forall x (x^2 \geq x)$ es: “el cuadrado de cualquier número entero x es mayor o igual al número entero x ”, ahora la negación de la expresión es: $\neg \forall x (x^2 \geq x) = \exists x \neg (x^2 \geq x) = \exists x (x^2 < x)$, donde el significado en palabras de la expresión $\neg \forall x (x^2 \geq x)$ es: “no es cierto que el cuadrado de cualquier número entero x es mayor o igual a número entero x ”. El significado en palabras de la expresión $\exists x \neg (x^2 \geq x)$ es: “existe algún número entero x para el cual no se cumple que el cuadrado del número entero x sea mayor o igual al número entero x ”. Por último, el significado en palabras de la expresión $\exists x (x^2 < x)$ es: “existe algún número entero x para el cual se cumple que el cuadrado del número entero x es menor al número entero x ”.

1.3.6. Ejercicios

1. Se definen los siguientes predicados con sus símbolos. Escriba cada enunciado simbólicamente. Hacer uso de los cuantificadores adecuados.

D(x): “x es un día”, S(x): “x es soleado”, L(x): “x es lluvioso”

- a) Todos los días son soleados
- b) Algunos días son lluviosos
- c) Todo día que es soleado no es lluvioso
- d) Algunos días son soleados y lluviosos
- e) Ningún día es soleado y lluvioso a la vez
- f) Solo es día soleado el día lluvioso

2. Si A(x): “x es un automóvil”, R(x): “x es rápido”, L(x): “x es lento”. Escriba cada enunciado simbólicamente. Atención a los cuantificadores.

- a) Algunos automóviles son rápidos y lentos.
- b) Algunos automóviles son rápidos.
- c) Ningún automóvil es lento y rápido a la vez.

- d) Todos los automóviles son lentos.
- e) Solo es automóvil rápido el automóvil lento.
- f) Todo automóvil que es rápido no es lento.

3. Escriba cada enunciado simbólicamente en la lógica de predicados.

- a) Todos los pájaros cantores vuelan.
- b) Algún pájaro cantor no vuela.
- c) No hay pájaros grandes que se alimenten de néctar.
- d) Perro no come perro.
- e) Hijo de tigre sale pintado.
- f) Algunos leones no toman café.
- g) Todos los peces, excepto los tiburones, son amables con los niños.
- h) Cualquier caballo que es manso está bien entrenado.
- i) Las serpientes son reptiles.
- j) Ningún automóvil que tenga más de diez años será reparado si está seriamente dañado.
- k) Ningún abrigo es impermeable a menos que haya sido especialmente tratado.
- l) Nadie sino los valientes merecen a la bella.
- m) Algunos senadores son o desleales o mal aconsejados.
- n) Sólo los ejecutivos tienen secretaria.
- \tilde{n}) Existe un entero que es mayor que 100 que es una potencia de 2.
- o) El sucesor de un número es un número.
- p) Para todo número natural, existe un número que es su inmediato sucesor.
- q) Para todo número natural diferente de cero, existe un número natural que es su inmediato predecesor.
- r) Todo número racional es un número real.
- s) Existe un número que es un primo.
- t) Para todo número x , existe un número y tal que $x < y$.
- u) Algunos números naturales son pares.
- v) Todo elemento $n \neq 1$ de \mathbb{N} es el siguiente de algún otro elemento de \mathbb{N} .
- w) Si $n, m \in \mathbb{N}$ y $m > n$ para todo $p \in \mathbb{N}$ entonces $m \cdot p > n \cdot p$

4. Sea $T(x, y)$: “ x es mas alto que y ”. El dominio consta de 5 estudiantes: Lina, que mide 155 cm, Federnam, que mide 170 cm, Francisco que mide 164 cm, Isabel, que mide 160 cm y Alexander, quien mide 168 cm. Escriba cada proposición con palabras e indique si es verdadera o falsa.

- a) $\forall_x \forall_y T(x, y)$.
- b) $\forall_x \exists_y T(x, y)$.
- c) $\exists_x \forall_y T(x, y)$.
- d) $\exists_x \exists_y T(x, y)$.
- e) $\forall_y \exists_x T(x, y)$.
- f) $\exists_x \forall_y T(y, x)$.

5. Si $A(x, y)$: “ x ama a y ”, $J(x)$: “ x es joven”, $H(x)$: “ x es un hombre”, $M(x)$: “ x es una mujer”, $B(x)$: “ x es bonita”, d : “Diego”, k : “Katherine”, Escriba el equivalente en español de los siguientes enunciados simbólicos:

- a) $(\forall x)(H(x) \rightarrow J(x))$
- b) $(\forall x)[M(x) \rightarrow (\forall y)(A(x, y) \rightarrow (H(y) \wedge J(y)))]$
- c) $(\exists x)(H(x) \wedge J(x) \wedge A(x, k))$
- d) $(\forall x)((M(x) \wedge B(x)) \rightarrow A(d, x))$

6. Sea $P(x, y)$: “ x es el padre de y ”, donde el dominio de x y y son los seres humanos del mundo.

- a) Representar el predicado $A(x, z)$: “ x es el abuelo de z ”, utilizando el predicado P y los cuantificadores si son necesarios.
- b) Representar el predicado $H(y, z)$: “ y es hermano de z ”, utilizando el predicado P y los cuantificadores si son necesarios.
- c) Representar el predicado $T(z, y)$: “ z es el tío de y ”, utilizando los predicados P y H (solución del item anterior) y los cuantificadores si son necesarios.

7. Sea el siguiente dominio $D = \{a, b\}$ para el predicado P que toma los siguientes valores de verdad para la combinación de valores del dominio:

$P(a, a)$	$P(a, b)$	$P(b, a)$	$P(b, b)$
V	F	F	V

Determine los valores de verdad para:

- a) $(\forall x)(\exists y)P(x, y)$
- b) $(\forall x)(\forall y)P(x, y)$
- c) $(\exists x)(\forall y)P(x, y)$
- d) $(\exists y) \sim P(x, y)$
- e) $(\forall x)P(x, x)$

8. Sean los siguientes predicados:

- $P(x) : x$ es primate
- $A(x) : x$ es arbóreo
- $R(x) : x$ es roedor

Escribir en palabras el significado de la siguiente expresión:

- a) $\exists x(P(x) \wedge A(x)) \wedge \exists x(R(x) \wedge A(x))$
- b) $\exists x(P(x) \wedge A(x)) \wedge \exists x((\sim R(x)) \wedge A(x))$
- c) $\forall x((P(x) \wedge A(x)) \rightarrow R(x))$
- d) $\forall x(R(x) \rightarrow (P(x) \vee A(x)))$

9. Sean los siguientes predicados:

- $O(x) : x$ es un organismo
- $M(x) : x$ es molusco
- $D(x) : x$ es doméstico

Escribir en palabras el significado de la siguiente expresión:

- a) $(\exists x(O(x) \wedge D(x))) \wedge (\exists x(O(x) \wedge M(x))) \wedge (\sim \exists x(O(x) \wedge D(x) \wedge M(x)))$
- b) $\forall x(M(x) \rightarrow O(x))$
- c) $\forall x(D(x) \rightarrow O(x))$
- d) $\sim \exists x((O(x) \wedge M(x)) \rightarrow D(x))$

1.4. Preguntas Tipo ECAES

1. El conectivo NOR (Not OR, No o) de la lógica matemática se representa con el símbolo \downarrow y su significado es la negación de una disyunción¹, es decir

$$(p \downarrow q) = \sim (p \vee q) = \overline{(p \vee q)}$$

La expresión $p \downarrow (q \downarrow r)$ es equivalente a:

- A.) $(p \downarrow q) \downarrow r$
 - B.) $\bar{p} \wedge (q \vee r)$
 - C.) $p \vee \overline{(q \vee r)}$
 - D.) $p \wedge \overline{(q \vee r)}$
 - E.) $\bar{p} \wedge (\bar{q} \wedge \bar{r})$
2. Para la proposición $(p \wedge \bar{r}) \wedge \overline{((q \vee r) \vee \overline{(p \vee r)})}$, ¿cuál de las siguientes opciones es la simplificación de la proposición?²

- A.) $p \wedge (\bar{q} \wedge \bar{r})$
 - B.) $\bar{p} \vee \overline{(q \vee r)}$
 - C.) $p \vee q$
 - D.) $(\bar{q} \wedge \bar{r}) \wedge (p \vee r)$
 - E.) $p \wedge \bar{r}$
3. Determine cuál de las siguientes proposiciones es lógicamente equivalente a:
- $$(p \wedge (q \vee r)) \rightarrow (r \vee q)$$

- A.) Contradicción
 - B.) $p \vee r$
 - C.) Tautología
 - D.) $p \wedge (q \vee r)$
 - E.) p
4. Identifique cuál de las siguientes proposiciones es lógicamente equivalente a:

$$\overline{(p \vee q)} \leftrightarrow \overline{(\bar{p} \wedge \bar{q})}$$

¹Pregunta reescrita a partir de una de las preguntas no liberadas del cuadernillo del ECAES de Ingeniería de Sistemas del año 2006.

²Las preguntas desde la 2 hasta la 9 han sido reescritas ajustándolas a las condiciones que tienen que cumplir las preguntas tipo ECAES en Colombia, a partir del material de auto evaluación de un libro mexicano de Matemáticas para la Computación de José A. Jiménez [J2009].

- A.) Tautología
- B.) $p \wedge q$
- C.) $p \vee q$
- D.) Contradicción
- E.) $\overline{(p \vee q)}$

5. Seleccione cuál de las siguientes proposiciones es lógicamente equivalente a:

$$\overline{(p \vee q)} \rightarrow \overline{(\bar{p} \wedge \bar{q})}$$

- A.) $p \wedge (q \vee \bar{r})$
- B.) $q \wedge (r \vee p)$
- C.) $\bar{p} \vee (q \wedge \bar{r})$
- D.) $p \vee q$
- E.) V_o

6. Identifique cuál de las siguientes proposiciones es lógicamente equivalente a:

$$((p \rightarrow \bar{s}) \wedge (p \rightarrow r))$$

- A.) $(r \leftrightarrow (\bar{p} \wedge s))$
- B.) $(\bar{p} \rightarrow (r \rightarrow \bar{s}))$
- C.) $(\bar{p} \vee (\bar{r} \wedge s))$
- D.) $(\bar{p} \wedge (r \vee \bar{s}))$
- E.) $(p \rightarrow (r \wedge \bar{s}))$

7. Seleccione cuál de las siguientes proposiciones es lógicamente equivalente a:

$$(((\bar{p} \rightarrow q) \wedge \bar{q}) \rightarrow p)$$

- A.) Contradicción
- B.) $(\bar{q} \rightarrow \bar{p})$
- C.) $(\bar{p} \wedge \bar{q}) \leftrightarrow (p \vee q)$
- D.) Tautología
- E.) p

8. Determine cuál de las siguientes proposiciones es lógicamente equivalente a:

$$\overline{((p \vee \bar{q}) \vee (\bar{p} \wedge q))}$$

- A.) $\bar{p} \leftrightarrow q$
- B.) Contradicción

C.) Tautología

D.) $\bar{p} \rightarrow \bar{q}$

E.) $\bar{p} \wedge q$

9. Identifique cuál de los siguientes argumentos es un argumento válido

A.) $((p \wedge q) \rightarrow r) \wedge (\bar{r} \vee s) \wedge (\bar{s}) \rightarrow (\bar{p} \vee \bar{q})$

B.) $((p \vee q) \wedge (p \rightarrow \bar{r})) \rightarrow (r \rightarrow (p \vee q))$

C.) $((p \rightarrow q) \wedge (q \rightarrow r)) \rightarrow (p \rightarrow \bar{r})$

D.) $((p \vee q) \wedge (q \rightarrow r) \wedge (p \rightarrow m) \wedge (\bar{m})) \rightarrow (r \wedge \bar{r})$

10. Sean los predicados $P(w, x)$: “ w es el padre de x ” y $M(y, z)$: “ y es madre de z ”, donde el dominio de w, x, y y z son los seres humanos del mundo. Adicionalmente, c denota a Camilo y d denota a Diego. ¿Cuál de las siguientes es la única representación correcta del enunciado: “Camilo es el abuelo paterno de Diego” ?³

A.) $\exists x \exists y (M(x, c) \wedge M(x, d) \wedge P(y, c) \wedge P(y, d))$

B.) $\exists x \exists y \exists z (P(x, y) \wedge P(x, z) \wedge M(y, c) \wedge P(z, d))$

C.) $\exists x (P(c, x) \wedge M(x, d))$

D.) $\exists x (P(c, x) \wedge P(x, d))$

11. Sean los predicados $P(w, x)$: “ w es el padre de x ” y $M(y, z)$: “ y es madre de z ”, donde el dominio de w, x, y y z son los seres humanos del mundo. Adicionalmente, c denota a Camilo y d denota a Diego. ¿Cuál de las siguientes es la única representación correcta del enunciado: “Camilo y Diego son hermanos” ?

A.) $\exists x \exists y (M(x, c) \wedge M(x, d) \wedge P(y, c) \wedge P(y, d))$

B.) $\exists x \exists y \exists z (P(x, y) \wedge P(x, z) \wedge M(y, c) \wedge P(z, d))$

C.) $\exists x (P(c, x) \wedge M(x, d))$

D.) $\exists x (P(c, x) \wedge P(x, d))$

12. Sean los predicados $P(w, x)$: “ w es el padre de x ” y $M(y, z)$: “ y es madre de z ”, donde el dominio de w, x, y y z son los seres humanos del mundo. Adicionalmente, c denota a Camilo y d denota a Diego. ¿Cuál de las siguientes es la única representación correcta del enunciado: “Camilo y Diego son primos hermanos” ?

A.) $\exists x \exists y (M(x, c) \wedge M(x, d) \wedge P(y, c) \wedge P(y, d))$

B.) $\exists x \exists y \exists z (P(x, y) \wedge P(x, z) \wedge M(y, c) \wedge P(z, d))$

C.) $\exists x (P(c, x) \wedge M(x, d))$

D.) $\exists x (P(c, x) \wedge P(x, d))$

³Las preguntas desde la 10 hasta la 17 son del autor.

13. Sean los predicados $P(w, x)$: “ w es el padre de x ” y $M(y, z)$: “ y es madre de z ”, donde el dominio de w, x, y y z son los seres humanos del mundo. Adicionalmente, c denota a Camilo y d denota a Diego. ¿Cuál de las siguientes es la única representación correcta del enunciado: “Camilo es el abuelo materno de Diego” ?
- A.) $\exists x \exists y (M(x, c) \wedge M(x, d) \wedge P(y, c) \wedge P(y, d))$
 B.) $\exists x \exists y \exists z (P(x, y) \wedge P(x, z) \wedge M(y, c) \wedge P(z, d))$
 C.) $\exists x (P(c, x) \wedge M(x, d))$
 D.) $\exists x (P(c, x) \wedge P(x, d))$
14. Sea el predicado $P(x, y)$: “ x es el padre de y ”, donde el dominio de x y y son los seres humanos del mundo. ¿Cuál de las siguientes es la única representación correcta del enunciado: “todo ser humano es hijo de algún otro ser humano” ?
- A.) $\forall x \exists y P(x, y)$
 B.) $\exists x \forall y P(x, y)$
 C.) $\forall x \forall y P(y, x)$
 D.) $\exists x \forall y P(y, x)$
 E.) $\forall x \exists y P(y, x)$
15. Sea el predicado $P(x, y)$: “ x es el padre de y ”, donde el dominio de x y y son los seres humanos del mundo. Utilizando el predicado P y los cuantificadores si son necesarios, ¿cuál de las siguientes es la única representación correcta para el predicado $A(x, z)$: “ x es el abuelo de z ”?
- A.) $A(x, z) : \forall x \forall y \forall z (P(x, y) \wedge P(y, z))$
 B.) $A(x, z) : \exists y (P(x, y) \wedge P(y, z))$
 C.) $A(x, z) : \forall y (P(x, y) \wedge P(y, z))$
 D.) $A(x, z) : \exists x \exists y \exists z (P(x, y) \wedge P(y, z))$
 E.) $A(x, z) : P(x, y) \wedge P(y, z)$
16. Sea el predicado $P(x, y)$: “ x es el padre de y ”, donde el dominio de x y y son los seres humanos del mundo. Utilizando el predicado P y los cuantificadores si son necesarios, ¿cuál de las siguientes es la única representación correcta para el predicado $H(y, z)$: “ y es hermano de z ” ?
- A.) $H(y, z) : \forall x \forall y \forall z (P(x, y) \wedge P(x, z))$
 B.) $H(y, z) : \exists x (P(x, y) \wedge P(x, z))$
 C.) $H(y, z) : \forall x (P(x, y) \wedge P(x, z))$
 D.) $H(y, z) : \exists x \exists y \exists z (P(x, y) \wedge P(x, z))$
 E.) $H(y, z) : P(x, y) \wedge P(x, z)$

17. Sean los predicados $P(w, x)$: “ w es el padre de x ” y $H(y, z)$: “ y es hermano de z ”, donde el dominio de w, x, y y z son los seres humanos del mundo.

Utilizando los predicados P y H y los cuantificadores si son necesarios, ¿cuál de las siguientes es la única representación correcta para el predicado $T(z, y)$: “ z es el tío de y ”?

- A.) $T(z, y) : \exists x(H(x, z) \wedge P(x, y))$
- B.) $T(z, y) : \forall x \exists y \forall z(H(x, z) \wedge P(x, y))$
- C.) $T(z, y) : \forall x \forall y \exists z(H(x, z) \wedge P(x, y))$
- D.) $T(z, y) : \exists x \exists y \exists z(H(x, z) \wedge P(x, y))$
- E.) $T(z, y) : H(x, y) \wedge P(x, y)$

18. En una tarjeta están escritas las siguientes 4 afirmaciones⁴ :

- En esta tarjeta hay exactamente una afirmación falsa
- En esta tarjeta hay exactamente dos afirmaciones falsas
- En esta tarjeta hay exactamente tres afirmaciones falsas
- En esta tarjeta hay exactamente cuatro afirmaciones falsas

El número de afirmaciones falsas en la tarjeta es:

- A.) 0
- B.) 1
- C.) 2
- D.) 3
- E.) 4

19. En una isla viven solo dos tipos de personas, los bribones y los caballeros. Los bribones siempre mienten y los caballeros siempre dicen la verdad. ¿Qué se puede concluir con respecto a la siguiente afirmación hecha por un habitante de la isla: “yo soy un bribón y mi compañero es un caballero”?⁵

- A.) El habitante que hizo la afirmación es un bribón y su compañero es un caballero.
- B.) El habitante que hizo la afirmación es un caballero y su compañero es un bribón.
- C.) El habitante que hizo la afirmación y su compañero son unos bribones.
- D.) El habitante que hizo la afirmación y su compañero son unos caballeros.
- E.) No hay suficiente información para dar una conclusión.

⁴Enunciado textual de la pregunta 14 del cuadernillo de la segunda sesión del ECAES de Ingeniería de Sistemas del año 2004 [MIA2004].

⁵Pregunta reescrita a partir de una de las preguntas no liberadas del cuadernillo de la primera sesión del ECAES de Ingeniería de Sistemas del año 2004.

Capítulo 2

Uso de sucesiones y sumatorias en computación

2.1. Funciones piso y techo

Definición: La función piso asigna al número real x el entero más grande que sea menor o igual a x . La función piso de x se denota por $\lfloor x \rfloor$.

Ejemplo 1:

$$\begin{array}{lll} \blacksquare \lfloor \frac{1}{2} \rfloor = 0 & \blacksquare \lfloor 3,00000001 \rfloor = 3 & \blacksquare \lfloor -3,78 \rfloor = -4 \\ \blacksquare \lfloor -\frac{1}{2} \rfloor = -1 & \blacksquare \lfloor 4,99999999 \rfloor = 4 & \blacksquare \lfloor 3,78 \rfloor = 3 \end{array}$$

Definición: La función techo asigna al número real x el entero más pequeño que sea mayor o igual a x . La función techo de x se denota por $\lceil x \rceil$.

Ejemplo 2:

$$\begin{array}{lll} \blacksquare \lceil \frac{1}{2} \rceil = 1 & \blacksquare \lceil 3,00000001 \rceil = 4 & \blacksquare \lceil -3,78 \rceil = -3 \\ \blacksquare \lceil -\frac{1}{2} \rceil = 0 & \blacksquare \lceil 4,99999999 \rceil = 5 & \blacksquare \lceil 3,78 \rceil = 4 \end{array}$$

2.1.1. Propiedades de las funciones piso y techo

En las siguientes propiedades $n \in \mathbb{Z}$ y $x \in \mathbb{R}$.

1. a.) $\lfloor x \rfloor = n$ si y solo si $n \leq x < n + 1$

- b.) $\lceil x \rceil = n$ si y solo si $n - 1 < x \leq n$
 c.) $\lfloor x \rfloor = n$ si y solo si $x - 1 < n \leq x$
 d.) $\lceil x \rceil = n$ si y solo si $x \leq n < x + 1$

2. $x - 1 < \lfloor x \rfloor \leq x \leq \lceil x \rceil < x + 1$

3. a.) $\lfloor -x \rfloor = -\lceil x \rceil$

b.) $\lceil -x \rceil = -\lfloor x \rfloor$

4. a.) $\lfloor x + n \rfloor = \lfloor x \rfloor + n$

b.) $\lceil x + n \rceil = \lceil x \rceil + n$

Ejemplo 3:

Probar que si x es un número real, entonces $\lfloor 2x \rfloor = \lfloor x \rfloor + \lfloor x + \frac{1}{2} \rfloor$.

Sea $x = n + \epsilon$, donde n representa la parte entera del número x y ϵ representa la parte decimal.

Se presentan dos casos, cuando $\epsilon < \frac{1}{2}$ y cuando $\epsilon \geq \frac{1}{2}$. En análisis de cada uno de los casos es el siguiente:

- Caso donde $\epsilon < \frac{1}{2}$:

$$\lfloor 2(n + \epsilon) \rfloor = \lfloor 2n + 2\epsilon \rfloor, \text{ como } 2\epsilon < 1, \text{ entonces } \lfloor 2n + 2\epsilon \rfloor = 2n$$

$$\lfloor n + \epsilon \rfloor + \left\lfloor n + \underbrace{\epsilon + \frac{1}{2}}_{\text{menor que } 1} \right\rfloor = n + n = 2n$$

$$\text{Por lo tanto } \lfloor 2x \rfloor = \lfloor x \rfloor + \lfloor x + \frac{1}{2} \rfloor = 2n$$

- Caso donde $\epsilon \geq \frac{1}{2}$:

$$\lfloor 2(n + \epsilon) \rfloor = \lfloor 2n + \underbrace{2\epsilon}_{\text{mayor o igual que } 1} \rfloor = 2n + 1$$

$$\lfloor n + \epsilon \rfloor + \left\lfloor n + \underbrace{\epsilon + \frac{1}{2}}_{\text{mayor o igual que } 1} \right\rfloor = n + n + 1 = 2n + 1$$

$$\text{Por lo tanto } \lfloor 2x \rfloor = \lfloor x \rfloor + \lfloor x + \frac{1}{2} \rfloor = 2n + 1$$

Como se cumplen los dos casos, entonces queda demostrado que $\lfloor 2x \rfloor = \lfloor x \rfloor + \lfloor x + \frac{1}{2} \rfloor$.

2.2. Sucesiones

Las sucesiones son usadas para representar listas ordenadas de elementos.

Definición: Una sucesión es una función del conjunto de los números naturales o del conjunto de los números enteros positivos a un conjunto S . Se usa la notación S_n para denotar la imagen del número n en el conjunto S , de esta misma forma, S_n representa al término ubicado en la posición n de la sucesión. Se usa la notación $\{S_n\}$ para denotar todo el rango de la función. Las sucesiones son descritas listando los términos de la sucesión en el orden como se va incrementando el subíndice.

Ejemplo 4:

Considerar la sucesión $\{S_n\}$, donde $S_n = \frac{n}{2^n}$.

Los elementos de la sucesión, comenzando en S_1 , son:

$$S_1 = \frac{1}{2^1}$$

$$S_2 = \frac{2}{2^2}$$

$$S_3 = \frac{3}{2^3}$$

$$S_4 = \frac{4}{2^4}$$

$$S_5 = \frac{5}{2^5}$$

$$\vdots$$

$$S_n = \frac{n}{2^n}$$

de esta forma se tiene que la lista de términos de la sucesión es:

$$\frac{1}{2^1}, \frac{2}{2^2}, \frac{3}{2^3}, \frac{4}{2^4}, \frac{5}{2^5}, \dots, \frac{n}{2^n}$$

Definición: Una progresión aritmética es una sucesión de la forma:

$$a, a + d, a + 2 \cdot d, a + 3 \cdot d, \dots, a + n \cdot d$$

donde el término inicial a y la diferencia constante d son números reales.

Ejemplo 5:

La sucesión $\{S_n\}$ con $S_n = 1 + 2 \cdot n$ es una progresión aritmética con término inicial $a = 1$ y diferencia constante $d = 2$.

Los elementos de la sucesión, comenzando en $n = 0$, son:

$$S_0 = 1 + 2 \cdot 0 = 1 + 0 = 1$$

$$S_1 = 1 + 2 \cdot 1 = 1 + 2 = 3$$

$$S_2 = 1 + 2 \cdot 2 = 1 + 4 = 5$$

$$S_3 = 1 + 2 \cdot 3 = 1 + 6 = 7$$

$$S_4 = 1 + 2 \cdot 4 = 1 + 8 = 9$$

\vdots

$$S_n = 1 + 2 \cdot n$$

de esta forma se tiene que la lista de términos de la sucesión es la lista de los números enteros positivos impares: 1, 3, 5, 7, 9, \dots , $1 + 2 \cdot n$.

Ejemplo 6:

Encontrar una fórmula para generar el n -ésimo término de la sucesión que tiene los primeros seis términos siguientes: 1, 5, 9, 13, 17 y 21.

Entre un par de términos consecutivos de la sucesión hay una diferencia constante de 4, se tiene como primer término de la sucesión al número 1, por este motivo la sucesión 1, 5, 9, 13, 17, 21, \dots , es una progresión aritmética con primer término $a = 1$ y diferencia constante $d = 4$.

La forma como se generan los primeros seis términos de la sucesión por medio de la progresión aritmética es:

$a + 0 \cdot d,$	$a + 1 \cdot d,$	$a + 2 \cdot d,$	$a + 3 \cdot d,$	$a + 4 \cdot d,$	$a + 5 \cdot d$
$1 + 0 \cdot 4,$	$1 + 1 \cdot 4,$	$1 + 2 \cdot 4,$	$1 + 3 \cdot 4,$	$1 + 4 \cdot 4,$	$1 + 5 \cdot 4$
$1 + 0,$	$1 + 4,$	$1 + 8,$	$1 + 12,$	$1 + 16,$	$1 + 20$
1,	5,	9,	13,	17,	21

Definición: Una progresión geométrica es una sucesión de la forma:

$$a, a \cdot r, a \cdot r^2, a \cdot r^3, \dots, a \cdot r^n$$

donde el término inicial a y la razón constante r son números reales.

Ejemplo 7:

La sucesión $\{S_n\}$ con $S_n = 2^n$ es una progresión geométrica con término inicial $a = 1$ y razón constante $r = 2$.

Los elementos de la sucesión, comenzando en $n = 0$, son:

$$S_0 = 2^0 = 1$$

$$S_1 = 2^1 = 2$$

$$\begin{aligned}
S_2 &= 2^2 = 4 \\
S_3 &= 2^3 = 8 \\
S_4 &= 2^4 = 16 \\
&\vdots \\
S_n &= 2^n
\end{aligned}$$

de esta forma se tiene que la lista de términos de la sucesión es:

$$1, 2, 4, 8, 16, \dots, 2^n$$

Ejemplo 8:

La sucesión $\{S_n\}$ con $S_n = 2 \cdot 3^n$ es una progresión geométrica con término inicial $a = 2$ y razón constante $r = 3$.

Los elementos de la sucesión, comenzando en $n = 0$, son:

$$\begin{aligned}
S_0 &= 2 \cdot 3^0 = 2 \cdot 1 = 2 \\
S_1 &= 2 \cdot 3^1 = 2 \cdot 3 = 6 \\
S_2 &= 2 \cdot 3^2 = 2 \cdot 9 = 18 \\
S_3 &= 2 \cdot 3^3 = 2 \cdot 27 = 54 \\
S_4 &= 2 \cdot 3^4 = 2 \cdot 81 = 162 \\
&\vdots \\
S_n &= 2 \cdot 3^n
\end{aligned}$$

de esta forma se tiene que la lista de términos de la sucesión es:

$$2, 6, 18, 54, 162, \dots, 2 \cdot 3^n$$

2.3. Sucesiones especiales de números

Cuando es difícil deducir una posible fórmula para generar el n -ésimo término de una sucesión, entonces, las siguientes son algunas de las preguntas que se deben formular:

- ¿La sucesión tiene siempre el mismo término?
- ¿Los términos son obtenidos de términos previos y la suma de alguna cantidad o los términos son obtenidos de una cantidad que depende de la posición en la sucesión?
- ¿Los términos son obtenidos de términos previos y la multiplicación de alguna cantidad o los términos son obtenidos de una cantidad que depende de la posición en la sucesión?
- ¿Los términos son obtenidos por la combinación de términos en una cierta forma?
- ¿Hay ciclos entre los términos de la sucesión?

Ejemplo 9:

Encontrar una fórmula para generar el n -ésimo término de la sucesión que tiene los primeros 10 términos siguientes: 3, 9, 15, 21, 27, 33, 39, 45, 51 y 57.

Para la sucesión de este ejemplo aplica la pregunta: “¿Los términos son obtenidos de términos previos y la suma de alguna cantidad o los términos son obtenidos de una cantidad que depende de la posición en la sucesión?”.

Con respecto a la primera opción de la pregunta la siguiente fórmula recursiva permite generar el n -ésimo término de la sucesión:

$$\begin{aligned} S_1 &= 3 \\ S_n &= S_{n-1} + 6, \text{ para } n \geq 2. \end{aligned}$$

Con respecto a la segunda opción de la pregunta se puede determinar que el n -ésimo término de la sucesión únicamente depende de su posición, ya que la sucesión es una progresión aritmética, donde el primer término es $a = 3$ y la diferencia común entre dos términos consecutivos de la sucesión es $d = 6$, por lo tanto la fórmula que genera el n -ésimo término de la sucesión es:

$$S_n = 3 + n \cdot 6, \text{ para } n \geq 0.$$

Ejemplo 10:

Encontrar una fórmula para generar el n -ésimo término de la sucesión que tiene los primeros cinco términos siguientes: 1, $\frac{2}{3}$, $\frac{4}{9}$, $\frac{8}{27}$ y $\frac{16}{81}$.

Para la sucesión de este ejemplo aplica la pregunta: “¿Los términos son obtenidos de términos previos y la multiplicación de alguna cantidad o los términos son obtenidos de una cantidad que depende de la posición en la sucesión?”.

Con respecto a la primera opción de la pregunta la siguiente fórmula recursiva permite generar el n -ésimo término de la sucesión:

$$\begin{aligned} S_1 &= 1 \\ S_n &= \frac{2}{3} \cdot S_{n-1}, \text{ para } n \geq 2. \end{aligned}$$

Con respecto a la segunda opción de la pregunta se puede determinar que el n -ésimo término de la sucesión únicamente depende de su posición, ya que la sucesión es una progresión geométrica, donde el primer término es $a = 1$ y la razón constante entre dos términos consecutivos de la sucesión es $r = \frac{2}{3}$, por lo tanto la fórmula que genera el n -ésimo término de la sucesión es:

$$S_n = 1 \cdot \left(\frac{2}{3}\right)^n = \left(\frac{2}{3}\right)^n, \text{ para } n \geq 0.$$

Ejemplo 11:

Encontrar una fórmula para generar el n -ésimo término de la sucesión que tiene los primeros seis términos siguientes: 1, 3, 6, 10, 15, 21, ...

Para la sucesión de este ejemplo aplica la pregunta: “¿Los términos son obtenidos de términos previos y la suma de alguna cantidad o los términos son obtenidos de una cantidad que depende de la posición en la sucesión?”.

Con respecto a la primera opción de la pregunta la siguiente fórmula recursiva permite generar el n -ésimo término de la sucesión:

$$S_1 = 1$$

$$S_n = S_{n-1} + n, \text{ para } n \geq 2.$$

Con respecto a la segunda opción de la pregunta se puede determinar que el n -ésimo término de la sucesión únicamente depende de su posición de la siguiente forma:

$$S_n = \frac{n(n+1)}{2}, \text{ para } n \geq 1.$$

Ejemplo 12:

Encontrar una fórmula para generar el n -ésimo término de la sucesión que tiene los primeros seis términos siguientes: 1, 1, 2, 6, 24, 120, ...

Para la sucesión de este ejemplo aplica la pregunta: “¿Los términos son obtenidos de términos previos y la multiplicación de alguna cantidad o los términos son obtenidos de una cantidad que depende de la posición en la sucesión?”.

Con respecto a la primera opción de la pregunta la siguiente fórmula recursiva permite generar el n -ésimo término de la sucesión:

$$S_0 = 1$$

$$S_n = n \cdot S_{n-1}, \text{ para } n \geq 1.$$

Con respecto a la segunda opción de la pregunta se puede determinar que el n -ésimo término de la sucesión únicamente depende de su posición de la siguiente forma:

$$S_n = n!, \text{ donde } n \geq 0$$

Ejemplo 13:

Encontrar una fórmula para generar el n -ésimo término de la sucesión que tiene los primeros 16 términos siguientes: 1, 2, 2, 2, 3, 3, 3, 3, 3, 4, 4, 4, 4, 4, 4 y 4.

Para la sucesión de este ejemplo aplica la segunda opción de la pregunta: “¿Los términos son obtenidos de términos previos y la suma de alguna cantidad o los términos son obtenidos de una cantidad que depende de la posición en la sucesión?”. Se puede

determinar que el n -ésimo término de la sucesión únicamente depende de su posición de la siguiente forma:

$$S_n = \lceil \sqrt{n} \rceil, \text{ para } n \geq 1.$$

Ejemplo 14:

Encontrar una fórmula para generar el n -ésimo término de la sucesión que tiene los primeros 15 términos siguientes: 1, 1, 1, 2, 2, 2, 2, 2, 3, 3, 3, 3, 3, 3 y 3.

Para la sucesión de este ejemplo aplica la segunda opción de la pregunta: “¿Los términos son obtenidos de términos previos y la suma de alguna cantidad o los términos son obtenidos de una cantidad que depende de la posición en la sucesión?”. Se puede determinar que el n -ésimo término de la sucesión únicamente depende de su posición de la siguiente forma:

$$S_n = \lfloor \sqrt{n} \rfloor, \text{ para } n \geq 1.$$

Ejemplo 15:

Encontrar una fórmula para generar el n -ésimo término de la sucesión que tiene los primeros 10 términos siguientes: 2, 7, 24, 77, 238, 723, 2180, 6553, 19674 y 59039.

Para la sucesión de este ejemplo aplica la pregunta: “¿Los términos son obtenidos por la combinación de términos en una cierta forma?”.

El n -ésimo término de la sucesión depende del resultado de una potencia de 3 menos una cantidad específica con respecto a la posición del término en la sucesión. La fórmula es la siguiente:

$$S_n = 3^n - n, \text{ para } n \geq 1.$$

Ejemplo 16:

Encontrar una fórmula para generar el n -ésimo término de la sucesión que tiene los primeros nueve términos siguientes: 0, 1, 1, 0, 1, 1, 0, 1, 1, ...

Para la sucesión de este ejemplo aplica la pregunta: “¿Hay ciclos entre los términos de la sucesión?”.

De forma cíclica los elementos 0, 1 y 1 se siguen presentado en la sucesión. Una fórmula que garantiza la generación cíclica de los términos es la siguiente:

$$S_0 = 0$$

$$S_1 = 1$$

$$S_2 = 1$$

$$S_n = S_{(n \bmod 3)}, \text{ para } n \geq 3.$$

Se debe tener en cuenta que $n \bmod 3$ es el residuo de la división entera entre n y 3, donde los únicos posibles residuos que se pueden obtener al dividir el número entero positivo n por 3 son: 0, 1 y 2.

2.4. Sumatorias

La notación de sumatoria es usada para representar la suma de los términos $S_m, S_{m+1}, S_{m+2}, \dots, S_n$ de la sucesión $\{S_n\}$.

Se usa la notación $\sum_{i=m}^n S_i$ para representar $S_m + S_{m+1} + S_{m+2} + \dots + S_n$.

Ejemplo 17:

La suma de los primeros 50 términos de la sucesión $\{S_n\}$ donde el n -ésimo término de la sucesión está definido por $S_n = \frac{n}{2^n}$ puede ser representada por medio de sumatorias como:

$$\sum_{i=1}^{50} \frac{i}{2^i} = \frac{1}{2^1} + \frac{2}{2^2} + \frac{3}{2^3} + \frac{4}{2^4} + \dots + \frac{50}{2^{50}}$$

Ejemplo 18:

¿Cuál es el resultado que se obtiene de $\sum_{i=1}^8 i$?

El resultado es el siguiente: $\sum_{i=1}^8 i = 1 + 2 + 3 + 4 + 5 + 6 + 7 + 8 = 36$.

Teorema:

$$\sum_{i=1}^n i = 1 + 2 + 3 + \dots + n = \frac{n(n+1)}{2}, \text{ para } n \in \mathbb{Z}^+.$$

Demostración:

Sea $S = 1 + 2 + 3 + \dots + n$, como el orden de los sumandos no altera el resultado entonces $S = n + (n-1) + (n-2) + \dots + 1$

$$\begin{array}{ccccccc} S = & 1 & + & 2 & + & 3 & + \dots + n \\ S = & n & + & (n-1) & + & (n-2) & + \dots + 1 \\ \hline 2S = & \underbrace{(n+1) + (n+1) + (n+1) + \dots + (n+1)}_{n \text{ veces}} \end{array}$$

$$2S = n(n+1)$$

$$S = \frac{n(n+1)}{2}$$

Como S es igual a $1 + 2 + 3 + \cdots + n$ entonces $1 + 2 + 3 + \cdots + n = \frac{n(n+1)}{2}$, para $n \in \mathbb{Z}^+$.

Ejemplo 19:

Obtener una fórmula que sea la solución de la siguiente suma de términos:

$2 + 4 + 6 + 8 + \cdots + 2 \cdot n$, para $n \in \mathbb{Z}^+$.

$$\begin{aligned} 2 + 4 + 6 + 8 + \cdots + 2 \cdot n &= 2 \cdot 1 + 2 \cdot 2 + 2 \cdot 3 + 2 \cdot 4 + \cdots + 2 \cdot n \\ &= 2 \cdot (1 + 2 + 3 + 4 + \cdots + n) \\ &= 2 \cdot \left(\sum_{i=1}^n i \right) \\ &= 2 \cdot \left(\frac{n(n+1)}{2} \right) \\ &= n(n+1) \end{aligned}$$

Por lo tanto $\sum_{i=1}^n 2 \cdot i = 2 + 4 + 6 + 8 + \cdots + 2 \cdot n = n(n+1)$

Ejemplo 20:

Obtener una fórmula que sea la solución de la siguiente suma de términos:

$1 + 3 + 5 + 7 + \cdots + 2n - 1$, para $n \in \mathbb{Z}^+$.

$$\begin{aligned} 1 + 3 + 5 + \cdots + 2 \cdot n - 1 &= 2 \cdot 1 - 1 + 2 \cdot 2 - 1 + 2 \cdot 3 - 1 + \cdots + 2 \cdot n - 1 \\ &= (2 \cdot 1 + 2 \cdot 2 + 2 \cdot 3 + \cdots + 2 \cdot n) - \underbrace{(1 + 1 + 1 + \cdots + 1)}_{n \text{ veces}} \\ &= 2 \cdot (1 + 2 + 3 + 4 + \cdots + n) - n \\ &= 2 \cdot \left(\sum_{i=1}^n i \right) - n \\ &= 2 \cdot \left(\frac{n(n+1)}{2} \right) - n \\ &= n(n+1) - n \\ &= n^2 + n - n \end{aligned}$$

$$= n^2$$

Por lo tanto $\sum_{i=1}^n (2 \cdot i - 1) = 1 + 3 + 5 + \cdots + 2 \cdot n - 1 = n^2$.

A continuación se presenta otra forma alternativa de obtener la solución de la suma de términos.

Sea $S = 1 + 3 + 5 + \cdots + 2n - 1$, como el orden de los sumandos no altera el resultado entonces $S = (2n - 1) + (2n - 3) + (2n - 5) + \cdots + 1$

$$\begin{array}{r} S = 1 + 3 + 5 + \cdots + 2n - 1 \\ S = (2n - 1) + (2n - 3) + (2n - 5) + \cdots + 1 \\ \hline 2S = \underbrace{(2n) + (2n) + (2n) + \cdots + (2n)}_{n \text{ veces}} \\ 2S = n(2n) \\ S = n^2 \end{array}$$

Como S es igual a $1 + 3 + 5 + \cdots + 2n - 1$ entonces $1 + 3 + 5 + \cdots + 2n - 1 = n^2$, para $n \in \mathbb{Z}^+$.

Teorema:

Si a y r son números reales con $r \neq 0$, entonces:

$$\sum_{i=0}^n a \cdot r^i = \begin{cases} a \cdot (n + 1) & \text{si } r = 1 \\ \frac{a \cdot r^{n+1} - a}{r - 1} & \text{si } r \neq 1 \end{cases}$$

Demostración:

Se debe demostrar cada uno de los dos casos del teorema de forma independiente, si los dos casos se cumplen entonces queda demostrada la validez del teorema

■ Caso donde $r = 1$

$$\sum_{i=0}^n a \cdot 1^i = \sum_{i=0}^n a \cdot 1 = \sum_{i=0}^n a = \underbrace{a + a + a + a + \cdots + a}_{n+1 \text{ veces}} = a \cdot (n + 1).$$

Queda demostrado el caso.

■ **Caso donde $r \neq 1$**

Sea $S = a + a \cdot r + a \cdot r^2 + a \cdot r^3 + \dots + a \cdot r^n$, al multiplicar a ambos lados de la igualdad por $-r$ se sigue conservando la igualdad, donde se obtiene: $-S \cdot r = -a \cdot r - a \cdot r^2 - a \cdot r^3 - a \cdot r^4 - \dots - a \cdot r^{n+1}$, al sumar ambas igualdades se tiene:

$$\begin{array}{rcl}
 S & = & a + a \cdot r + a \cdot r^2 + a \cdot r^3 + \dots + a \cdot r^n \\
 -S \cdot r & = & -a \cdot r - a \cdot r^2 - a \cdot r^3 - a \cdot r^4 - \dots - a \cdot r^{n+1} \\
 \hline
 S - S \cdot r & = & a - a \cdot r^{n+1}
 \end{array}$$

$$S \cdot (1 - r) = a - a \cdot r^{n+1}$$

$$S = \frac{a - a \cdot r^{n+1}}{1 - r}$$

$$S = \frac{-1}{-1} \cdot \frac{a - a \cdot r^{n+1}}{1 - r}$$

$$S = \frac{-a + a \cdot r^{n+1}}{-1 + r}$$

$$S = \frac{a \cdot r^{n+1} - a}{r - 1}$$

Como S es igual a $a + a \cdot r + a \cdot r^2 + a \cdot r^3 + \dots + a \cdot r^n$ entonces $a + a \cdot r + a \cdot r^2 + a \cdot r^3 + \dots + a \cdot r^n = \frac{a \cdot r^{n+1} - a}{r - 1}$, para $n \in \mathbb{Z}^+$.

Queda demostrado el caso.

Como se cumplen todos los casos del teorema entonces queda demostrada la validez del teorema de la suma de términos de la serie o progresión geométrica.

Ejemplo 21:

Obtener una fórmula que sea la solución de la siguiente suma de términos:

$$2^0 - 2^1 + 2^2 - 2^3 + 2^4 + \dots + (-1)^n \cdot 2^n, \text{ donde } n \in \mathbb{N}.$$

Los términos que se están sumando pertenecen a una serie geométrica, con primer término $a = 2^0 = 1$, razón constante $r = -2$ y potencia más grande igual a n , al reemplazar los valores en la fórmula que es la solución de la sumatoria de términos tenemos:

$$2^0 - 2^1 + 2^2 - 2^3 + 2^4 + \dots + (-1)^n \cdot 2^n = \frac{1 \cdot (-2)^{n+1} - 1}{-2 - 1} = \frac{(-2)^{n+1} - 1}{-3}$$

Ejemplo 22:

Obtener una fórmula que sea la solución de la siguiente suma de términos:

$5^4 + 5^5 + 5^6 + \dots + 5^n$, donde $n \in \mathbb{Z}^+$.

Los términos que se están sumando pertenecen a una serie geométrica, pero la fórmula que representa la solución de la suma términos no se puede utilizar directamente porque la potencia más pequeña no comienza en cero, por este motivo es necesario realizar el siguiente trabajo:

$$\begin{aligned}
 5^4 + 5^5 + 5^6 + \dots + 5^n &= 5^{4+0} + 5^{4+1} + 5^{4+2} + \dots + 5^{4+(n-4)} \\
 &= 5^4 \cdot 5^0 + 5^4 \cdot 5^1 + 5^4 \cdot 5^2 + \dots + 5^4 \cdot 5^{n-4} \\
 &= 5^4 \cdot \underbrace{(5^0 + 5^1 + 5^2 + \dots + 5^{n-4})}_{\substack{\text{Serie geométrica con } a = 1, r = 5 \\ \text{y potencia más grande } n - 4}} \\
 &= 5^4 \cdot \left(\frac{1 \cdot 5^{n-4+1} - 1}{5 - 1} \right) \\
 &= 5^4 \cdot \left(\frac{5^{n-3} - 1}{4} \right)
 \end{aligned}$$

Otra forma de resolver el ejercicio haciendo uso de sumatorias es la siguiente:

$$\begin{aligned}
 5^4 + 5^5 + 5^6 + \dots + 5^n &= ((5^0 + 5^1 + 5^2 + 5^3) + (5^4 + 5^5 + \dots + 5^n)) - (5^0 + 5^1 + 5^2 + 5^3) \\
 \underbrace{5^4 + 5^5 + 5^6 + \dots + 5^n}_{\sum_{i=4}^n 5^i} &= \underbrace{(5^0 + 5^1 + 5^2 + 5^3 + 5^4 + 5^5 + \dots + 5^n)}_{\sum_{i=0}^n 5^i} - \underbrace{(5^0 + 5^1 + 5^2 + 5^3)}_{\sum_{i=0}^3 5^i} \\
 &= \frac{1 \cdot 5^{n+1} - 1}{5 - 1} - \frac{1 \cdot 5^4 - 1}{5 - 1} \\
 &= \frac{5^{n+1} - 1}{4} - \frac{5^4 - 1}{4} \\
 &= \frac{5^{n+1} - 1 - 5^4 + 1}{4} \\
 &= \frac{5^{n+1} - 5^4}{4} \\
 &= 5^4 \cdot \left(\frac{5^{n-3} - 1}{4} \right)
 \end{aligned}$$

Las dos formas de resolver el ejercicio (ya se por factorización o por complemento y sustracción de los términos que faltan en la serie geométrica) permiten obtener el mismo

resultado.

Ejemplo 23:

Obtener una fórmula que sea la solución de la siguiente suma de términos¹ :

$$\sum_{i=1}^n \frac{i}{2^i} = \frac{1}{2^1} + \frac{2}{2^2} + \frac{3}{2^3} + \cdots + \frac{n}{2^n}, \text{ para } n \in \mathbb{Z}^+.$$

Los términos que se están sumando no pertenecen originalmente a una serie geométrica, pero se puede hacer un manejo de los términos para que la solución de la sumatoria de términos de la serie geométrica pueda ser utilizada. El manejo es el siguiente:

$$\begin{aligned} \sum_{i=1}^n \frac{i}{2^i} &= \frac{1}{2^1} + \frac{2}{2^2} + \frac{3}{2^3} + \cdots + \frac{n}{2^n} \\ &= \frac{1}{2^1} + \left(\frac{1}{2^2} + \frac{1}{2^2} \right) + \left(\frac{1}{2^3} + \frac{1}{2^3} + \frac{1}{2^3} \right) + \cdots + \underbrace{\left(\frac{1}{2^n} + \frac{1}{2^n} + \frac{1}{2^n} + \cdots + \frac{1}{2^n} \right)}_{n \text{ veces}} \\ &= \left(\frac{1}{2^1} + \frac{1}{2^2} + \frac{1}{2^3} + \cdots + \frac{1}{2^n} \right) + \left(\frac{1}{2^2} + \frac{1}{2^3} + \cdots + \frac{1}{2^n} \right) + \left(\frac{1}{2^3} + \frac{1}{2^4} + \cdots + \frac{1}{2^n} \right) \\ &\quad + \cdots + \left(\frac{1}{2^n} \right) \\ &= \sum_{i=1}^n \left(\frac{1}{2} \right)^i + \sum_{i=2}^n \left(\frac{1}{2} \right)^i + \sum_{i=3}^n \left(\frac{1}{2} \right)^i + \cdots + \sum_{i=n}^n \left(\frac{1}{2} \right)^i \\ &= \left(\sum_{i=0}^n \left(\frac{1}{2} \right)^i - \sum_{i=0}^0 \left(\frac{1}{2} \right)^i \right) + \left(\sum_{i=0}^n \left(\frac{1}{2} \right)^i - \sum_{i=0}^1 \left(\frac{1}{2} \right)^i \right) + \left(\sum_{i=0}^n \left(\frac{1}{2} \right)^i - \sum_{i=0}^2 \left(\frac{1}{2} \right)^i \right) \\ &\quad + \cdots + \left(\sum_{i=0}^n \left(\frac{1}{2} \right)^i - \sum_{i=0}^{n-1} \left(\frac{1}{2} \right)^i \right) \\ &= n \left(\sum_{i=0}^n \left(\frac{1}{2} \right)^i \right) - \left(\sum_{i=0}^0 \left(\frac{1}{2} \right)^i + \sum_{i=0}^1 \left(\frac{1}{2} \right)^i + \sum_{i=0}^2 \left(\frac{1}{2} \right)^i + \cdots + \sum_{i=0}^{n-1} \left(\frac{1}{2} \right)^i \right) \\ &= n \left(\frac{\left(\frac{1}{2} \right)^{n+1} - 1}{\frac{1}{2} - 1} \right) - \left(\frac{\left(\frac{1}{2} \right)^1 - 1}{\frac{1}{2} - 1} + \frac{\left(\frac{1}{2} \right)^2 - 1}{\frac{1}{2} - 1} + \frac{\left(\frac{1}{2} \right)^3 - 1}{\frac{1}{2} - 1} + \cdots + \frac{\left(\frac{1}{2} \right)^n - 1}{\frac{1}{2} - 1} \right) \\ &= n \left(\frac{\frac{1}{2^{n+1}} - 1}{-\frac{1}{2}} \right) - \left(\frac{\left(\frac{1}{2} \right)^1 - 1}{-\frac{1}{2}} + \frac{\left(\frac{1}{2} \right)^2 - 1}{-\frac{1}{2}} + \frac{\left(\frac{1}{2} \right)^3 - 1}{-\frac{1}{2}} + \cdots + \frac{\left(\frac{1}{2} \right)^n - 1}{-\frac{1}{2}} \right) \\ &= n \left(\frac{\frac{1}{2^{n+1}} - 1}{-\frac{1}{2}} \right) - \left(\frac{\left(\frac{1}{2} \right)^1 - 1 + \left(\frac{1}{2} \right)^2 - 1 + \left(\frac{1}{2} \right)^3 - 1 + \cdots + \left(\frac{1}{2} \right)^n - 1}{-\frac{1}{2}} \right) \end{aligned}$$

¹El enunciado de este ejercicio y la solución han sido tomadas del “Anexo A” de [M2010].

$$\begin{aligned}
&= n \left(\frac{\frac{-1+2^{n+1}}{2^{n+1}}}{\frac{1}{2}} \right) + 2 \left(\left(\left(\frac{1}{2} \right)^1 + \left(\frac{1}{2} \right)^2 + \left(\frac{1}{2} \right)^3 + \cdots + \left(\frac{1}{2} \right)^n \right) - n \right) \\
&= n \left(\frac{-1+2^{n+1}}{2^n} \right) + 2 \left(\left(\sum_{i=1}^n \left(\frac{1}{2} \right)^i \right) - n \right) \\
&= n \left(\frac{-1+2^{n+1}}{2^n} \right) - 2n + 2 \sum_{i=1}^n \left(\frac{1}{2} \right)^i \\
&= n \left(\frac{-1}{2^n} + \frac{2^{n+1}}{2^n} \right) - 2n + 2 \left(\sum_{i=0}^n \left(\frac{1}{2} \right)^i - \sum_{i=0}^0 \left(\frac{1}{2} \right)^i \right) \\
&= n \left(\frac{-1}{2^n} + 2 \right) - 2n + 2 \left(\frac{\left(\frac{1}{2} \right)^{n+1} - 1}{\frac{1}{2} - 1} - \frac{\left(\frac{1}{2} \right)^1 - 1}{\frac{1}{2} - 1} \right) \\
&= -\frac{n}{2^n} + 2n - 2n + 2 \left(\frac{\frac{1}{2^{n+1}} - \frac{2^{n+1}}{2^{n+1}}}{-\frac{1}{2}} - \frac{\frac{1}{2} - 1}{\frac{1}{2} - 1} \right) \\
&= -\frac{n}{2^n} + 2 \left(\frac{\frac{-1+2^{n+1}}{2^{n+1}}}{\frac{1}{2}} - 1 \right) \\
&= -\frac{n}{2^n} + 2 \left(\frac{-1+2^{n+1}}{2^n} - 1 \right) \\
&= -\frac{n}{2^n} + 2 \left(-\frac{1}{2^n} + \frac{2^{n+1}}{2^n} - 1 \right) \\
&= -\frac{n}{2^n} + 2 \left(-\frac{1}{2^n} + 2 - 1 \right) \\
&= -\frac{n}{2^n} + 2 \left(-\frac{1}{2^n} + 1 \right) \\
&= -\frac{n}{2^n} - \frac{2}{2^n} + 2 \\
&= 2 - \frac{n}{2^n} - \frac{2}{2^n} \\
&= 2 - \frac{1}{2^n} (n+2) \\
&= 2 - \frac{n+2}{2^n}
\end{aligned}$$

Por lo tanto

$$\sum_{i=1}^n \frac{i}{2^i} = \frac{1}{2^1} + \frac{2}{2^2} + \frac{3}{2^3} + \cdots + \frac{n}{2^n} = 2 - \frac{n+2}{2^n}, \text{ para } n \in \mathbb{Z}^+.$$

2.4.1. Fórmulas de sumatorias útiles:

Las siguientes son algunas de las sumatorias más importantes (o más utilizadas) en Matemáticas Computacionales junto con su solución. Esta información será de gran importancia en el Capítulo 4 de Relaciones de Recurrencia, cuando se trabaje en la sección 4.1 el Método de Iteración

- $\sum_{i=1}^n i = \frac{n}{2}(n+1)$, para $n \in \mathbb{Z}^+$.
- $\sum_{i=1}^n i^2 = \frac{n(n+1)(2n+1)}{6}$, para $n \in \mathbb{Z}^+$.
- $\sum_{i=1}^n i^3 = \frac{n^2(n+1)^2}{4}$, para $n \in \mathbb{Z}^+$.
- $\sum_{i=1}^n i^4 = \frac{n(n+1)(2n+1)(3n^2+3n-1)}{30}$, para $n \in \mathbb{Z}^+$.
- $\sum_{i=0}^n a \cdot r^i = \frac{ar^{n+1} - a}{r - 1}$, donde $r \neq 0$, $r \neq 1$ y $n \in \mathbb{N}$.
- Si $|r| < 1$ y $r \neq 0$, entonces $\sum_{i=0}^{\infty} a \cdot r^i = \frac{a}{1-r}$, para $i \in \mathbb{N}$.
- $\sum_{i=1}^n i \cdot 2^i = (n-1)2^{n+1} + 2$, para $n \in \mathbb{Z}^+$.
- $\sum_{i=1}^n \frac{i}{2^i} = 2 - \frac{n+2}{2^n}$, para $n \in \mathbb{Z}^+$.

Ejemplo 24:

Obtener una fórmula que sea la solución de la siguiente suma de términos:

$$1 \cdot \frac{1}{5} + 2 \cdot \frac{16}{5} + 3 \cdot \frac{41}{5} + \dots + n \cdot \left(n^2 - \frac{4}{5}\right), \text{ donde } n \in \mathbb{Z}^+.$$

Haciendo uso de sumatorias la solución es la siguiente:

$$\begin{aligned} \sum_{i=1}^n i \cdot \left(i^2 - \frac{4}{5}\right) &= 1 \cdot \left(1^2 - \frac{4}{5}\right) + 2 \cdot \left(2^2 - \frac{4}{5}\right) + 3 \cdot \left(3^2 - \frac{4}{5}\right) + \dots + n \cdot \left(n^2 - \frac{4}{5}\right) \\ &= 1^3 - \frac{4}{5} \cdot 1 + 2^3 - \frac{4}{5} \cdot 2 + 3^3 - \frac{4}{5} \cdot 3 + \dots + n^3 - \frac{4}{5} \cdot n \\ &= \left(1^3 + 2^3 + 3^3 + \dots + n^3\right) - \frac{4}{5} \left(1 + 2 + 3 + \dots + n\right) \\ &= \left(\sum_{i=1}^n i^3\right) - \frac{4}{5} \cdot \left(\sum_{i=1}^n i\right) \end{aligned}$$

$$\begin{aligned}
&= \frac{n^2 \cdot (n+1)^2}{4} - \frac{4}{5} \cdot \left(\frac{n \cdot (n+1)}{2} \right) \\
&= \frac{n \cdot (n+1)}{2} \cdot \left[\frac{n \cdot (n+1)}{2} - \frac{4}{5} \right] \\
&= \frac{n \cdot (n+1)}{2} \cdot \left[\frac{5 \cdot n \cdot (n+1) - 8}{10} \right] \\
&= \frac{n \cdot (n+1) \cdot (5 \cdot n \cdot (n+1) - 8)}{20}
\end{aligned}$$

Por lo tanto $\sum_{i=1}^n i \cdot \left(i^2 - \frac{4}{5} \right) = \frac{n \cdot (n+1) \cdot (5 \cdot n \cdot (n+1) - 8)}{20}$, para $n \in \mathbb{Z}^+$.

Ejemplo 25:

Obtener una fórmula que sea la solución de la siguiente suma de términos:

$1^2 + 3^2 + 5^2 + 7^2 + \dots + (2n-1)^2$?, donde $n \in \mathbb{Z}^+$.

Haciendo uso de sumatorias la solución es la siguiente:

$$\begin{aligned}
\sum_{i=1}^n (2 \cdot i - 1)^2 &= \sum_{i=1}^n ((2 \cdot i)^2 + 2 \cdot (2 \cdot i) \cdot (-1) + (-1)^2) \\
&= \sum_{i=1}^n (4 \cdot i^2 - 4 \cdot i + 1) \\
&= \left(\sum_{i=1}^n 4 \cdot i^2 \right) - \left(\sum_{i=1}^n 4 \cdot i \right) + \left(\sum_{i=1}^n 1 \right) \\
&= 4 \cdot \left(\sum_{i=1}^n i^2 \right) - 4 \cdot \left(\sum_{i=1}^n i \right) + n \\
&= 4 \cdot \left(\frac{n \cdot (n+1) \cdot (2n+1)}{6} \right) - 4 \cdot \left(\frac{n \cdot (n+1)}{2} \right) + n \\
&= 2 \cdot \left(\frac{n \cdot (n+1) \cdot (2n+1)}{3} \right) - 2 \cdot (n \cdot (n+1)) + n \\
&= \left(\frac{2 \cdot n \cdot (n+1)}{3} \right) \cdot ((2n+1) - 3) + n \\
&= \left(\frac{2 \cdot n \cdot (n+1)}{3} \right) \cdot (2n - 2) + n \\
&= \left(\frac{2 \cdot n \cdot (n+1) \cdot 2 \cdot (n-1)}{3} \right) + n
\end{aligned}$$

$$\begin{aligned}
&= \left(\frac{4 \cdot n \cdot (n+1) \cdot (n-1)}{3} \right) + n \\
&= \left(\frac{n}{3} \right) \cdot \left(4 \cdot (n+1) \cdot (n-1) + 3 \right) \\
&= \left(\frac{n}{3} \right) \cdot \left(4 \cdot (n^2 - 1) + 3 \right) \\
&= \left(\frac{n}{3} \right) \cdot \left(4 \cdot n^2 - 4 + 3 \right) \\
&= \left(\frac{n}{3} \right) \cdot \left(4 \cdot n^2 - 1 \right) \\
&= \left(\frac{n}{3} \right) \cdot \left((2 \cdot n - 1) \cdot (2 \cdot n + 1) \right) \\
&= \frac{n \cdot (2n - 1) \cdot (2n + 1)}{3}
\end{aligned}$$

De esta forma se obtiene que $1^2 + 3^2 + 5^2 + 7^2 + \dots + (2n-1)^2 = \frac{n \cdot (2n-1) \cdot (2n+1)}{3}$, para $n \in \mathbb{Z}^+$.

2.5. Sumatorias dobles y su utilización en ciclos anidados

Ejemplo 26:

¿Qué valor devuelve la siguiente función?. Expresar la respuesta en términos de n .

función Misterio(n : Entero Positivo)

$r = 0$

Para $i = 1$ Hasta $n - 1$

Para $j = i + 1$ Hasta n

Para $k = 1$ Hasta j

$r = r + 1$

Retornar r

$$Misterio(n) = \sum_{i=1}^{n-1} \sum_{j=i+1}^n \sum_{k=1}^j 1$$

$$= \sum_{j=2}^n \sum_{k=1}^j 1 + \sum_{j=3}^n \sum_{k=1}^j 1 + \dots + \sum_{j=n}^n \sum_{k=1}^j 1$$

$$\begin{aligned}
&= \left(\sum_{k=1}^2 1 + \sum_{k=1}^3 1 + \dots + \sum_{k=1}^n 1 \right) + \left(\sum_{k=1}^3 1 + \sum_{k=1}^4 1 + \dots + \sum_{k=1}^n 1 \right) + \dots + \\
&\quad \left(\sum_{k=1}^{n-1} 1 + \sum_{k=1}^n 1 \right) + \left(\sum_{k=1}^n 1 \right) \\
&= (2 + 3 + \dots + n) + (3 + 4 + \dots + n) + \dots + ((n-1) + n) + (n) \\
&= \left(\sum_{k=2}^n 1 \right) + \left(\sum_{k=3}^n 1 \right) + \dots + \left(\sum_{k=n-1}^n 1 \right) + \left(\sum_{k=n}^n 1 \right) \\
&= \left(\sum_{k=1}^n 1 - \sum_{k=1}^1 1 \right) + \left(\sum_{k=1}^n 1 - \sum_{k=1}^2 1 \right) + \dots + \left(\sum_{k=1}^n 1 - \sum_{k=1}^{n-2} 1 \right) + \\
&\quad \left(\sum_{k=1}^n 1 - \sum_{k=1}^{n-1} 1 \right) \\
&= \left(\frac{n \cdot (n+1)}{2} - \frac{1 \cdot 2}{2} \right) + \left(\frac{n \cdot (n+1)}{2} - \frac{2 \cdot 3}{2} \right) + \dots + \\
&\quad \left(\frac{n \cdot (n+1)}{2} - \frac{(n-2) \cdot (n-1)}{2} \right) + \left(\frac{n \cdot (n+1)}{2} - \frac{(n-1) \cdot n}{2} \right) \\
&= (n-1) \cdot \left(\frac{n \cdot (n+1)}{2} \right) - \left(\frac{1 \cdot 2}{2} + \frac{2 \cdot 3}{2} + \frac{3 \cdot 4}{2} + \dots + \frac{(n-1) \cdot (n)}{2} \right) \\
&= \frac{(n-1) \cdot n \cdot (n+1)}{2} - \frac{1}{2} \cdot \left(\sum_{i=1}^{n-1} i \cdot (i+1) \right) \\
&= \frac{(n-1) \cdot n \cdot (n+1)}{2} - \frac{1}{2} \cdot \left(\sum_{i=1}^{n-1} (i^2 + 1) \right) \\
&= \frac{(n-1) \cdot n \cdot (n+1)}{2} - \frac{1}{2} \cdot \left(\sum_{i=1}^{n-1} i^2 + \sum_{i=1}^{n-1} 1 \right) \\
&= \frac{(n-1) \cdot n \cdot (n+1)}{2} - \frac{1}{2} \cdot \left(\frac{(n-1) \cdot n \cdot (2 \cdot (n-1) + 1)}{6} + \frac{(n-1) \cdot n}{2} \right) \\
&= \frac{(n-1) \cdot n \cdot (n+1)}{2} - \frac{(n-1) \cdot n \cdot (2 \cdot n - 2 + 1)}{12} - \frac{(n-1) \cdot n}{4} \\
&= \frac{6 \cdot (n-1) \cdot n \cdot (n+1)}{6 \cdot 2} - \frac{(n-1) \cdot n \cdot (2 \cdot n - 1)}{6 \cdot 2} - \frac{3 \cdot (n-1) \cdot (n)}{3 \cdot 4} \\
&= \frac{(n-1) \cdot n}{12} \cdot \left(6 \cdot (n+1) - (2n-1) - 3 \right)
\end{aligned}$$

$$\begin{aligned}
&= \frac{(n-1) \cdot n}{12} \cdot (6n + 6 - 2n + 1 - 3) \\
&= \frac{(n-1) \cdot n}{12} \cdot (4n + 4) \\
&= \frac{(n-1) \cdot n \cdot (4 \cdot (n+1))}{4 \cdot 3} \\
&= \frac{(n-1) \cdot n \cdot (n+1)}{3}
\end{aligned}$$

Ejemplo 27:

¿Qué valor devuelve la siguiente función?. Expresar la respuesta en términos de n .

función Campeón(n : Entero Positivo)

$r = 0$

Para $i = 0$ Hasta $n - 1$

Para $j = i + 1$ Hasta n

$r = r + j + 2^i$

Retornar r

$$\text{Campeón}(n) = \sum_{i=0}^{n-1} \sum_{j=i+1}^n (j + 2^i) = \underbrace{\sum_{i=0}^{n-1} \sum_{j=i+1}^n j}_{\text{Sumatoria A}} + \underbrace{\sum_{i=0}^{n-1} \sum_{j=i+1}^n 2^i}_{\text{Sumatoria B}}$$

Sumatoria A :

$$\begin{aligned}
\sum_{i=0}^{n-1} \sum_{j=i+1}^n j &= \sum_{j=1}^n j + \sum_{j=2}^n j + \dots + \sum_{j=n}^n j \\
&= \sum_{j=1}^n j + \left(\sum_{j=1}^n j - \sum_{j=1}^1 j \right) + \left(\sum_{j=1}^n j - \sum_{j=1}^2 j \right) + \dots + \left(\sum_{j=1}^n j - \sum_{j=1}^{n-1} j \right) \\
&= n \cdot \left(\sum_{j=1}^n j \right) - \left(\sum_{j=1}^1 j + \sum_{j=1}^2 j + \dots + \sum_{j=1}^{n-1} j \right) \\
&= n \cdot \left(\frac{n \cdot (n+1)}{2} \right) - \left(\frac{1 \cdot 2}{2} + \frac{2 \cdot 3}{2} + \frac{3 \cdot 4}{2} + \dots + \frac{(n-1) \cdot n}{2} \right) \\
&= \frac{n^2 \cdot (n+1)}{2} - \frac{1}{2} \cdot \left(1 \cdot 2 + 2 \cdot 3 + 3 \cdot 4 + \dots + (n-1) \cdot n \right) \\
&= \frac{n^2 \cdot (n+1)}{2} - \frac{1}{2} \cdot \left(\sum_{i=1}^{n-1} i \cdot (i+1) \right)
\end{aligned}$$

$$\begin{aligned}
&= \frac{n^2 \cdot (n+1)}{2} - \frac{1}{2} \cdot \left(\sum_{i=1}^{n-1} (i^2 + i) \right) \\
&= \frac{n^2 \cdot (n+1)}{2} - \frac{1}{2} \cdot \left(\sum_{i=1}^{n-1} i^2 + \sum_{i=1}^{n-1} i \right) \\
&= \frac{n^2 \cdot (n+1)}{2} - \frac{1}{2} \cdot \left(\frac{(n-1) \cdot n \cdot (2 \cdot (n-1) + 1)}{6} + \frac{(n-1) \cdot n}{2} \right) \\
&= \frac{n^2 \cdot (n+1)}{2} - \frac{1}{2} \cdot \left(\frac{(n-1) \cdot n \cdot (2n-1)}{2 \cdot 3} + \frac{(n-1) \cdot n}{2} \right) \\
&= \frac{n^2 \cdot (n+1)}{2} - \frac{1}{2} \cdot \left(\frac{(n-1) \cdot n}{2} \cdot \left(\frac{2n-1}{3} + 1 \right) \right) \\
&= \frac{n^2 \cdot (n+1)}{2} - \frac{1}{2} \cdot \left(\frac{(n-1) \cdot n}{2} \cdot \left(\frac{2n-1+3}{3} \right) \right) \\
&= \frac{n^2 \cdot (n+1)}{2} - \frac{1}{2} \cdot \left(\frac{(n-1) \cdot n}{2} \cdot \left(\frac{2n+2}{3} \right) \right) \\
&= \frac{n^2 \cdot (n+1)}{2} - \frac{1}{2} \cdot \left(\frac{(n-1) \cdot n}{2} \cdot \left(\frac{2 \cdot (n+1)}{3} \right) \right) \\
&= \frac{n^2 \cdot (n+1)}{2} - \frac{1}{2} \cdot \left(\frac{(n-1) \cdot n \cdot (n+1)}{3} \right) \\
&= \frac{n \cdot (n+1)}{2} \cdot \left(n - \frac{n-1}{3} \right) \\
&= \frac{n \cdot (n+1)}{2} \cdot \left(\frac{3n - n + 1}{3} \right) \\
&= \frac{n \cdot (n+1)}{2} \cdot \left(\frac{2n+1}{3} \right) \\
&= \frac{n \cdot (n+1) \cdot (2n+1)}{6}
\end{aligned}$$

Sumatoria B :

$$\begin{aligned}
\sum_{i=0}^{n-1} \sum_{j=i+1}^n 2^i &= \sum_{j=1}^n 2^0 + \sum_{j=2}^n 2^1 + \dots + \sum_{j=n}^n 2^n \\
&= n \cdot 2^0 + (n-1) \cdot 2^1 + (n-2) \cdot 2^2 + \dots + (n-(n-1)) \cdot 2^{n-1} \\
&= n \cdot 2^0 + n \cdot 2^1 - 1 \cdot 2^1 + n \cdot 2^2 - 2 \cdot 2^2 + \dots + n \cdot 2^{n-1} - (n-1) \cdot 2^{n-1} \\
&= (n \cdot 2^0 + n \cdot 2^1 + n \cdot 2^2 + \dots + n \cdot 2^{n-1}) - (1 \cdot 2^1 + 2 \cdot 2^2 + \dots + (n-1) \cdot 2^{n-1}) \\
&= n \cdot (2^0 + 2^1 + 2^2 + \dots + 2^{n-1}) - (1 \cdot 2^1 + 2 \cdot 2^2 + \dots + (n-1) \cdot 2^{n-1})
\end{aligned}$$

$$\begin{aligned}
&= n \cdot \left(\sum_{i=0}^{n-1} 2^i \right) - \left(\sum_{i=1}^{n-1} i \cdot 2^i \right) \\
&= n \cdot \left(\frac{2^{n-1+1} - 1}{2 - 1} \right) - ((n-1-1) \cdot 2^{n-1+1} + 2) \\
&= n \cdot (2^n - 1) - ((n-2) \cdot 2^n + 2) \\
&= n \cdot 2^n - n - n \cdot 2^n + 2 \cdot 2^n - 2 \\
&= 2 \cdot 2^n - 2 - n \\
&= 2 \cdot (2^n - 1) - n
\end{aligned}$$

$$\begin{aligned}
\text{Campeón}(n) &= (\text{Solución Sumatoria A}) + (\text{Solución Sumatoria B}) \\
&= \frac{n \cdot (n+1) \cdot (2n+1)}{6} + 2 \cdot (2^n - 1) - n
\end{aligned}$$

2.6. Notaciones computacionales (O, Omega, Theta)

El tiempo requerido para solucionar un problema no depende solamente del número de operaciones que realiza, el tiempo de ejecución también depende del software y hardware usado para ejecutar el programa en el que se implementa el algoritmo. Sin embargo cuando se cambia el software o hardware se debe aproximar el tiempo requerido para solucionar un problema de tamaño n , multiplicando el tiempo previo por una constante.

El trabajo con notaciones computacionales (ya sea O , Ω o Θ) permite estimar el crecimiento de la función de costo de un algoritmo (dicho en otras palabras, permite calcular la complejidad computacional de un algoritmo), sin necesidad de preocuparse por el software o hardware que se utilice para implementarlo.

Se asumirá que las diferentes operaciones (como las de suma, multiplicación, división, resta, asignación, comparación, etc.) usadas en un algoritmo toman el mismo tiempo, lo cual simplifica el análisis de la complejidad del algoritmo considerablemente.

2.6.1. Notación O

La notación O es usada para estimar el número máximo de operaciones que un algoritmo realiza a partir de sus datos de entrada. Esta notación permite calcular la cota superior para una función $f(n)$, para valores de $n \geq k$. $f(n)$ es la función de costo del algoritmo, función que representa la cantidad de operaciones que realiza el algoritmo para los datos con los que trabaja.

Definición de O :

Sean f y g funciones desde el conjunto de los números reales al conjunto de los números reales. Se dice que $f(n)$ es $O(g(n))$ si hay constantes positivas c y k tales que:

$$|f(n)| \leq c \cdot |g(n)|, \text{ donde } n \geq k.$$

k es un número natural a partir del cual se comienza a cumplir el orden $O(g(n))$ y c es una constante que pertenece a los números reales positivos.

Nota 1: Las constantes c y k son llamadas testigos para que la función $f(n)$ sea $O(g(n))$.

Nota 2: No hay un único par de testigos c y k para la función $f(n)$, hay infinitos testigos y donde lo único que importa es encontrar al menos un par de estos.

Ejemplo 28:

¿Cuál es la notación O de $f(n) = 1 + 2 + 3 + \dots + n$?

Cada uno de los términos que se están sumando es menor o igual a n , por este motivo es válida la siguiente forma de aplicar la notación O :

$$f(n) = 1 + 2 + 3 + \dots + n \leq \underbrace{n + n + n + \dots + n}_{n \text{ veces}} = n \cdot n = n^2 = \underbrace{1}_c \cdot \underbrace{n^2}_{g(n)}$$

Con lo anterior se tiene que $f(n) = 1 + 2 + 3 + \dots + n$ es $O(n^2)$ donde los testigos son $c = 1$ y $k = 1$.

La forma ideal de aplicar la notación O es a partir de la solución de la suma de términos, donde tenemos:

$$f(n) = 1 + 2 + 3 + \dots + n = \frac{n \cdot (n + 1)}{2} = \frac{n^2}{2} + \frac{n}{2} \leq \frac{n^2}{2} + \frac{n^2}{2} = \underbrace{1}_c \cdot \underbrace{n^2}_{g(n)}$$

De esta forma $f(n) = 1 + 2 + \dots + n$ es $O(n^2)$ donde los testigos son $c = 1$ y $k = 1$.

En este ejemplo se obtiene la misma cota superior para la función $f(n)$ independientemente de la variante que se puede aplicar para obtener ésta, ya sea a partir del reemplazo de uno a uno de los términos que se están sumando por el más grande, o a partir de la fórmula que es la solución de la sumatoria de términos. Pero esto no siempre será así, en el próximo ejemplo las notaciones O que se obtienen no son la misma.

Ejemplo 29:

¿Cuál es la notación O para $f(n) = 2^0 + 2^1 + 2^2 + \dots + 2^n$?

El término 2^n es el más grande de los que se están sumando. Al aplicar la notación O término a término en la función $f(n)$ con respecto a éste valor, tenemos:

$$\begin{aligned} \underbrace{2^0 + 2^1 + 2^2 + \dots + 2^n}_{f(n)} &\leq \underbrace{2^n + 2^n + 2^n + \dots + 2^n}_{(n+1) \text{ veces}} = (n+1)2^n = n \cdot 2^n + 2^n \\ &\leq n \cdot 2^n + n \cdot 2^n = \underbrace{2}_c \cdot \underbrace{n \cdot 2^n}_{g(n)} \end{aligned}$$

Con el análisis anterior se tiene que $f(n) = 2^0 + 2^1 + \dots + 2^n$ es $O(n \cdot 2^n)$ con $c = 2$ y $k = 1$ como testigos. Obsérvese que en este caso el testigo $k = 0$ no sirve porque no se cumpliría la definición de la notación O , obteniéndose $1 \leq 0$, lo cual es falso.

A partir de la solución de la suma de términos tenemos:

$$f(n) = 2^0 + 2^1 + 2^2 + \dots + 2^n = 2^{n+1} - 1 \leq 2^{n+1} = \underbrace{2}_c \cdot \underbrace{2^n}_{g(n)}$$

Con este análisis la función $f(n) = 2^0 + 2^1 + \dots + 2^n$ es $O(2^n)$ con testigos $c = 2$ y $k = 0$.

Las diferentes variantes para aplicar la notación O no siempre dan el mismo resultado, previamente se obtuvo que la función $f(n)$ es acotada superiormente por $n \cdot 2^n$ y por 2^n , donde 2^n acota mucho más cercanamente a la función $f(n)$ versus $n \cdot 2^n$. Desde el punto de vista computacional, siempre se va a preferir la función que acote más cercanamente a la función $f(n)$. Las mejores cotas superiores se obtienen al aplicar la notación O sobre la fórmula que es la solución de la sumatoria de términos, cuando esto sea posible.

2.6.2. Notación Ω

La notación Omega (Ω) es usada para estimar el número mínimo de operaciones que un algoritmo realiza a partir de sus datos de entrada. Esta notación permite calcular la cota inferior para una función $f(n)$, para valores de $n \geq k$. $f(n)$ es la función de costo del algoritmo, función que representa la cantidad de operaciones que realiza el algoritmo para los datos con los que trabaja.

Definición

Sean f y g funciones desde el conjunto de los números reales al conjunto de los números reales, Se dice que $f(n)$ es $\Omega(g(n))$ si hay constantes positivas c y k tales que:

$$|f(n)| \geq c \cdot |g(n)| \quad \text{donde } n \geq k.$$

k es un número natural a partir del cual se comienza a cumplir el orden $\Omega(g(n))$ y c es una constante que pertenece a los números reales positivos.

Ejemplo 30:

¿Cuál es la mejor notación Ω para $f(n) = 1 + 2 + 3 + \dots + n$?

Al trabajar directamente sobre los términos que se están sumando tenemos:

$$1 + 2 + 3 + 4 + \dots + n \geq n = \underbrace{1}_c \cdot \underbrace{n}_{g(n)}$$

De esta forma se puede garantizar que la función $f(n) = 1 + 2 + 3 + \dots + n$ es $\Omega(n)$ con testigos $c = 1$ y $k = 1$, pero, ésta no es la mejor cota inferior.

Al trabajar sobre la solución de la sumatoria de términos tenemos:

$$f(n) = 1 + 2 + 3 + \dots + n = \frac{n \cdot (n+1)}{2} = \frac{n^2}{2} + \frac{n}{2} \geq \frac{n^2}{2} = \underbrace{\frac{1}{2}}_c \cdot \underbrace{n^2}_{g(n)}$$

La función $f(n) = 1 + 2 + 3 + \dots + n$ es $\Omega(n^2)$ donde se tienen los testigos $c = \frac{1}{2}$ y $k = 1$.

La mejor notación Ω para la función $f(n) = 1 + 2 + 3 + \dots + n$ es n^2 porque es la cota inferior más cercana. Desde el punto de vista computacional, siempre se le va a dar prioridad a la cota más cercana a la función.

Ejemplo 31:

¿Cuál es la notación Ω para $f(n) = 2^0 + 2^1 + 2^2 + \dots + 2^n$?

Como se puede garantizar que la mejor notación ya sea O u Ω se obtiene al trabajar sobre el resultado de la solución de la sumatoria de términos, entonces se presenta a continuación únicamente el análisis sobre dicha solución:

$$f(n) = 2^0 + 2^1 + 2^2 + \dots + 2^n = 2^{n+1} - 1 \geq \frac{2^{n+1}}{2} = \underbrace{1}_c \cdot \underbrace{2^n}_{g(n)}$$

La función $f(n) = 2^0 + 2^1 + 2^2 + \dots + 2^n$ es $\Omega(2^n)$ con $c = 1$ y $k = 0$ como testigos.

2.6.3. Notación Θ

La notación Theta (Θ) permite definir tanto una cota superior e inferior para la función $f(n)$.

Definición:

Sean f y g funciones desde el conjunto de los números reales al conjunto de los números reales. Se dice que $f(n)$ es $\Theta(g(n))$ si $f(n)$ es al mismo tiempo $O(g(n))$ y $\Omega(g(n))$. Dicho en otras palabras, $f(n)$ es $\Theta(g(n))$ si la misma función $g(n)$ acota superior e inferiormente a la función $f(n)$.

Cuando $f(n)$ es $\Theta(g(n))$ se dice que $f(n)$ es de orden $g(n)$.

Ejemplo 32:

¿Cuál es la notación Θ de $f(n) = 1 + 2 + 3 + \dots + n$?

En ejemplos anteriores ya se hizo el análisis para la función $f(n)$ con respecto a las notaciones O y Ω , donde se obtuvo la misma función $g(n) = n^2$, por este motivo se garantiza que la suma de los primeros n enteros positivos es $\Theta(n^2)$, o dicho en otras palabras es de orden n^2 .

Ejemplo 33:

¿Cuál es la notación Θ para la suma de términos $\frac{1}{2^1} + \frac{2}{2^2} + \frac{3}{2^3} + \frac{4}{2^4} + \dots + \frac{n}{2^n}$?

Se tiene que $f(n) = \sum_{i=1}^n \frac{i}{2^i} = \frac{1}{2^1} + \frac{2}{2^2} + \frac{3}{2^3} + \frac{4}{2^4} + \dots + \frac{n}{2^n} = 2 - \frac{n+2}{2^n}$, para $n \in \mathbb{Z}^+$.

Análisis de la notación O :

$$f(n) = 2 - \frac{n+2}{2^n} \leq 2 = \underbrace{2}_c \cdot \underbrace{1}_{g(n)}$$

de esta forma $f(n) = \sum_{i=1}^n \frac{i}{2^i}$ es $O(1)$ con testigos $c = 2$ y $k = 1$.

Análisis de la notación Ω :

$$f(n) = 2 - \frac{n+2}{2^n} \geq \frac{1}{2} = \underbrace{\frac{1}{2}}_c \cdot \underbrace{1}_{g(n)}$$

de esta forma $f(n) = \sum_{i=1}^n \frac{i}{2^i}$ es $\Omega(1)$ con testigos $c = \frac{1}{2}$ y $k = 1$.

Se obtuvo la misma función $g(n) = 1$ tanto en la notación O como en la notación Ω , por este motivo $f(n) = \frac{1}{2^1} + \frac{2}{2^2} + \frac{3}{2^3} + \frac{4}{2^4} + \dots + \frac{n}{2^n}$ es $\Theta(1)$.

2.7. Complejidad de algoritmos iterativos

En esta sección se presenta el análisis de la complejidad de dos algoritmos clásicos de ordenamiento por comparaciones, el ordenamiento por selección y el ordenamiento por inserción.

2.7.1. El Algoritmo de Ordenamiento por Selección

La idea en la que se apoya el algoritmo de ordenamiento por selección, es, recorrer el vector seleccionando la posición en la cual se encuentra el elemento más pequeño almacenado en este, el elemento seleccionado es intercambiado con el que se encuentre en la primera posición del vector, luego en el subvector que falta por ordenar se selecciona la posición del elemento más pequeño para luego ser intercambiado con el elemento que se encuentra en la segunda posición del vector, y así sucesivamente hasta que queda ordenado todo el vector. El pseudocódigo del algoritmo de ordenamiento por selección se presenta a continuación.

SelectionSort(A)

1. for $i \leftarrow 1$ to $length[A] - 1$
2. do $minInd \leftarrow i$
3. for $j \leftarrow i + 1$ to $length[A]$
4. do if $A[j] < A[minInd]$
5. then $minInd \leftarrow j$
6. $A[minInd] \leftrightarrow A[i]$

En la siguiente tabla se presenta el valor de las variables i , $minInd$ y de todas las posiciones del vector A después de ejecutarse la línea 6 del algoritmo, para cuando se le pide a éste ordenar la información del vector $A = \langle 5, 1, 3, 2, 4, 6, 1, 3 \rangle$, la cual es presentada en la primera fila de la tabla.

i	minInd	A[1]	A[2]	A[3]	A[4]	A[5]	A[6]	A[7]	A[8]
-	-	5	1	3	2	4	6	1	3
1	2	1	5	3	2	4	6	1	3
2	7	1	1	3	2	4	6	5	3
3	4	1	1	2	3	4	6	5	3
4	4	1	1	2	3	4	6	5	3
5	8	1	1	2	3	3	6	5	4
6	8	1	1	2	3	3	4	5	6
7	7	1	1	2	3	3	4	5	6

En la tabla, para cada una de las filas, se han encerrado en un recuadro los elementos que se han intercambiado. Cuando el elemento más pequeño del subvector se encuentra en la posición a intercambiar no es necesario realizar intercambio (o si se realiza esto no modifica en nada el vector), esto se ve reflejado en aquellas filas donde hay un solo elemento dentro de un recuadro.

Si ya se tiene clara la forma como trabaja el algoritmo entonces el análisis de la complejidad puede ser el siguiente, para el elemento ubicado en la primera posición es necesario realizar $n - 1$ comparaciones para seleccionar el elemento más pequeño a quedar allí, para el elemento ubicado en la segunda posición es necesario realizar $n - 2$ comparaciones para seleccionar el elemento más pequeño a quedar allí, y así sucesivamente hasta el elemento ubicado en la posición $n - 1$ del vector donde es necesario realizar una sola comparación para seleccionar el elemento de los dos últimos a quedar allí. Sea $T(n)$ la función que cuenta el total de comparaciones que realiza el algoritmo de ordenamiento por selección, $T(n) = (n-1) + (n-2) + \dots + 2 + 1$, $T(n) = 1 + 2 + 3 \dots + (n-1) = \frac{(n-1) \cdot n}{2}$.

Aplicando notación O :

$$T(n) = \frac{(n-1) \cdot n}{2} = \frac{n^2 - n}{2} = \frac{n^2}{2} - \frac{n}{2} \leq \frac{n^2}{2} = \underbrace{\frac{1}{2}}_c \cdot \underbrace{n^2}_{g(n)}, \text{ para } n \geq k = 1.$$

por lo tanto el algoritmo de ordenamiento por selección es $O(n^2)$.

Aplicando notación Ω :

$$T(n) = \frac{(n-1) \cdot n}{2} = \frac{n^2}{2} - \frac{n}{2} \geq \frac{n^2}{4} = \underbrace{\frac{1}{4}}_c \cdot \underbrace{n^2}_{g(n)}, \text{ para } n \geq k = 2$$

de esta forma el algoritmo de ordenamiento por selección es $\Omega(n^2)$.

Aplicando notación Θ :

El algoritmo de ordenamiento por selección es $\Theta(n^2)$ o dicho de otra forma es de orden n^2 , debido a que el algoritmo tiene el mismo $g(n) = n^2$ en las notaciones O y Ω .

2.7.2. El Algoritmo de Ordenamiento por Inserción

Toda esta subsección ha sido tomada de [M2010], donde dicha referencia tomo como punto de partida al libro de *Introducción a los Algoritmos* de Thomas Cormen y otros [CLRS2001].

El algoritmo Insertion Sort es un algoritmo eficiente para ordenar una cantidad pequeña de elementos. El algoritmo Insertion Sort trabaja de la forma como muchas personas ordenan una mano de cartas de poker, donde se comienza con la mano izquierda vacía y las cartas boca abajo sobre la mesa, se toma una a una las cartas de la mesa y se inserta

en la posición correcta en las cartas que ya están ordenadas en la mano izquierda. Para encontrar la posición correcta de una carta, esta se compara con cada una de las cartas que ya están ordenadas en la mano izquierda, de derecha a izquierda, la siguiente figura (tomada de [CLRS2001]) ilustra la situación planteada



El pseudo código del algoritmo Insertion Sort es el siguiente:

```

InsertionSort(A)
1.  for  $j \leftarrow 2$  to  $length[A]$ 
2.      do  $key \leftarrow A[j]$ 
3.           $i \leftarrow j - 1$ 
4.          while  $i > 0$  and  $A[i] > key$ 
5.              do  $A[i + 1] \leftarrow A[i]$ 
6.                   $i \leftarrow i - 1$ 
7.           $A[i + 1] \leftarrow key$ 

```

La complejidad del Insertion Sort en el *mejor de los casos*, caso que se presenta cuando recibe un arreglo ordenado de forma ascendente es $\theta(n)$, esto se presenta porque el algoritmo nunca ejecuta en este caso el ciclo *while* de las líneas 4-6 porque nunca se cumple la condición de éste ciclo de repetición, de esta forma únicamente se recorren las n posiciones del arreglo para confirmar que la información de cada casilla del arreglo está en la posición donde tiene que quedar.

La complejidad del Insertion Sort en el *peor de los casos*, caso que se presenta cuando recibe un arreglo ordenado de forma descendente es $\theta(n^2)$, esto se presenta porque el algoritmo ejecuta el ciclo *while* para desplazar a la derecha todos los elementos que se encuentran en posiciones inferiores a la posición que se está analizando, de ésta forma el ciclo *while* se ejecuta una vez para el elemento de la posición dos, se ejecuta dos veces para el elemento de la posición tres, se ejecuta tres veces para el elemento de la posición cuatro, y así sucesivamente hasta llegar a la posición n donde el ciclo se tiene que ejecutar $n - 1$ veces, de ésta forma el total de veces que se ejecuta el ciclo *while* es $1 + 2 + 3 + \dots + n - 1 = \frac{(n-1) \cdot n}{2}$, $\frac{(n-1) \cdot n}{2} = \frac{n^2 - n}{2} = \frac{n^2}{2} - \frac{n}{2} = \theta(n^2)$.

La complejidad del Insertion Sort en el *caso promedio* es la misma que se presenta en el Insertion Sort en el peor de los casos, donde se tiene que el ciclo *while* se ejecuta un orden de $\theta(n^2)$ veces.

2.8. La programación recursiva y su costo computacional

Los estudiantes de Ingeniería de Sistemas deben manejar adecuadamente las definiciones recursivas por medio del manejo de funciones de esta característica, y la programación de éstas, dentro del paradigma de programación que utilice. Normalmente se espera que el paradigma de programación funcional sea el que se utilice para programar funciones recursivas, pero esto no es una camisa de fuerza: por ejemplo, Lenguaje C es un lenguaje de programación que pertenece al paradigma imperativo que permite programar normalmente funciones recursivas, lo mismo sucede con el lenguaje de programación JAVA el cual pertenece al paradigma orientado a objetos y donde se pueden programar sin ningún problema este tipo de funciones.

¿Por qué un estudiante de Ingeniería de Sistemas debe hacer buen uso de las funciones recursivas?

- Porque muchos problemas matemáticos se definen de forma natural y sencilla de forma recursiva.
- Algunas de las estructuras de datos más utilizada en los lenguajes de programación, son de naturaleza recursiva, tal es el caso de las listas, pilas, colas y árboles.
- Al resolver un problema de optimización por medio de programación dinámica, primero se debe plantear una solución al problema de forma recursiva, para luego ser implementada sin recursividad, calculando cada uno de los subproblemas la primera vez que se hace referencia a ellos, y para todas las otras referencias a estos subproblemas se toma el valor ya previamente calculado, sin necesidad de volver a repetir este proceso.

Es importante que los estudiantes de Ingeniería de Sistemas no sólo interpreten correctamente las definiciones y funciones recursivas, sino que también, deben de tener la habilidad de resolver diferentes tipos de problemas de forma eficiente por medio de esta característica. Adicionalmente, es importante acabar con el “mito” o creencia en los estudiantes que la programación recursiva es costosa computacionalmente, y que por lo tanto no vale la pena aprender a programar de ésta forma. Realmente, quien hace esta afirmación es porque tiene un desconocimiento total del como se programa correcta y eficientemente de manera recursiva.

Así como en los algoritmos iterativos las sumatorias y notaciones computacionales sirven para la determinar la complejidad de un algoritmo de este tipo, en programación

recursiva se hace uso tanto de sumatorias, como de relaciones de recurrencia y notaciones computacionales para calcular la complejidad en este tipo de algoritmos.

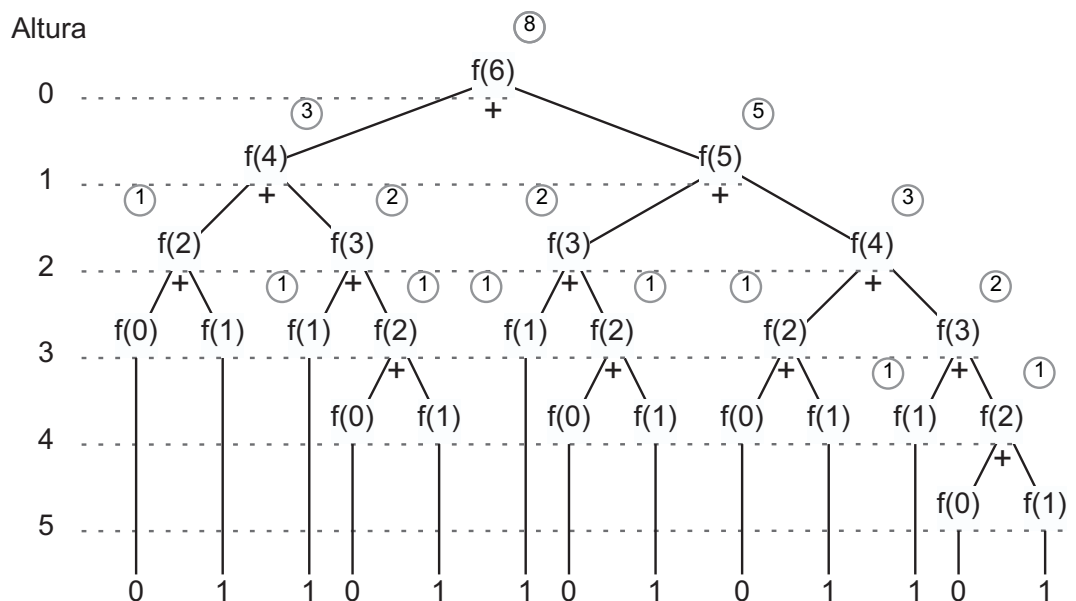
A continuación se presenta el análisis de la complejidad de la función recursiva que permite generar el n -ésimo término de la sucesión de Fibonacci.

2.8.1. La sucesión de Fibonacci

La “sucesión de Fibonacci” es representada por: 0, 1, 1, 2, 3, 5, 8, 13, 21, ..., donde el n -ésimo término es generado por la suma de los términos que se encuentran en las posiciones $n - 2$ y $n - 1$ en la sucesión, a excepción de los dos primeros los cuales no cumplen con esta regla de generación y se dan como punto de partida de la sucesión. La siguiente es una función recursiva que permite generar el n -ésimo término de la sucesión, teniendo en cuenta las reglas de generación anterior:

$$f(n) = \begin{cases} 0 & \text{si } n = 0 \\ 1 & \text{si } n = 1 \\ f(n-2) + f(n-1) & \text{si } n > 1 \end{cases}$$

El siguiente es el árbol de llamados recursivos que se genera para calcular el elemento que está en la posición número seis de la sucesión de Fibonacci:



Notación Omega de Fibonacci recursivo tradicional

A partir del árbol de llamados recursivos se puede observar que el árbol es completo hasta la altura $\lfloor \frac{n}{2} \rfloor$, la cual es la longitud de la rama más corta (rama que se encuentra

ubicada en el extremo izquierdo). En nuestro ejemplo sería $\lfloor \frac{6}{2} \rfloor = 3$. El total de llamados recursivos hasta el nivel que está completo es:

$$2^0 + 2^1 + 2^2 + \dots + 2^{\lfloor \frac{n}{2} \rfloor} = 2^{\lfloor \frac{n}{2} \rfloor + 1} - 1 \geq 2^{\lfloor \frac{n}{2} \rfloor} = \underbrace{1}_c \cdot \underbrace{2^{\lfloor \frac{n}{2} \rfloor}}_{g(n)}, \text{ para } n \geq k = 1$$

con lo anterior se garantiza que el Fibonacci recursivo tradicional es $\Omega(2^{\lfloor \frac{n}{2} \rfloor})$ con testigos $c = 1$ y $k = 1$.

La notación Ω del Fibonacci recursivo tradicional, garantiza que como mínimo la cantidad de llamados que se realizan de la función f es exponencial, para cuando ésta es evaluada en un valor n .

Ejemplo 34:

Según la notación theta del Fibonacci recursivo tradicional, ¿cuántos llamados se hacen a la función f cuando esta se evalúa en un valor de $n = 50$?

Como el Fibonacci recursivo tradicional es $\Omega(2^{\lfloor \frac{n}{2} \rfloor})$, entonces, una cota inferior que representa la cantidad mínima de llamados que se realizan de la función f cuando es evaluada en 50, es: $2^{\lfloor \frac{50}{2} \rfloor} = 2^{\lfloor 25 \rfloor} = 2^{25} = 67'108,864$.

Notación O de Fibonacci recursivo tradicional

En el árbol de llamados recursivos se puede observar que el árbol no es totalmente balanceado hasta su altura (rama que se encuentra ubicada en el extremo derecho), pero, si se considera que el árbol de llamados recursivos fuese completo hasta su altura (longitud de la rama más larga), el análisis de la notación O de la cantidad de llamados de la función f es:

$$2^0 + 2^1 + 2^2 + \dots + 2^{n-1} = 2^n - 1 \leq 2^n = \underbrace{1}_c \cdot \underbrace{2^n}_{g(n)}, \text{ para } n \geq k = 0$$

con lo anterior se garantiza que el Fibonacci recursivo tradicional es $O(2^n)$ con testigos $c = 1$ y $k = 0$.

La notación O del Fibonacci recursivo tradicional, garantiza que como máximo la cantidad de llamados que se realizan de la función f es exponencial, para cuando ésta es evaluada en un valor n .

Ejemplo 35:

Según la notación O del Fibonacci recursivo tradicional, ¿cuántos llamados se hacen a la función f cuando ésta se evalúa en un valor de $n = 50$?

Como el Fibonacci recursivo tradicional es $O(2^n)$, entonces, una cota superior que representa la cantidad máxima de llamados que se realizan de la función f cuando es evaluada en 50, es: $2^{50} = 1,125'899,906'842,624$.

Notación Theta de Fibonacci recursivo tradicional

A partir de la estrategia que se tomó para calcular las notaciones Ω y O , no se obtuvo una misma función $g(n)$, por ese motivo y por el momento, no se puede concluir una notación Θ para la función recursiva tradicional de Fibonacci, pero, a partir de las cotas inferior y superior se puede afirmar lo siguiente:

$$2^{\lfloor \frac{n}{2} \rfloor} < \text{Total de llamados de la función } f \text{ al evaluarse en } n < 2^n$$

2.8.2. Fibonacci mejorado

Tomando de nuevo la sucesión de Fibonacci: 0, 1, 1, 2, 3, 5, 8, 13, 21, ..., se puede identificar una nueva forma recursiva que tome como punto de partida los valores iniciales de la sucesión (0 y 1) y a partir de ellos, en una función auxiliar g , genere el siguiente valor de la sucesión, en este caso el 1 ($0+1 = 1$), luego la función auxiliar g trabajará sobre los valores (1, 1) y generará el 2 ($1 + 1 = 2$), y así sucesivamente seguirá trabajando la función auxiliar g recibiendo los valores de la sucesión que le permiten generar el número siguiente en la sucesión, esto se repetirá hasta que se alcance el n -ésimo valor solicitado en la sucesión.

A continuación se presenta la función f_m que hace uso de la función recursiva auxiliar g , donde toman forma las ideas anteriormente planteadas:

$$f_m(n) = \begin{cases} 0 & \text{si } n = 0 \\ 1 & \text{si } n = 1 \\ g(n, 2, 0, 1) & \text{si } n > 1 \end{cases}$$

$$g(n, i, a, b) = \begin{cases} a + b & \text{si } i = n \\ g(n, i + 1, b, a + b) & \text{si } i < n \end{cases}$$

La generación de los números de Fibonacci a partir de la función mejorada f_m para las posiciones 0, 1, 2, 3, 4, 5 y 6, es:

- $f_m(0) = 0$
- $f_m(1) = 1$
- $f_m(2) = g(2, 2, 0, 1) = 0 + 1 = 1$
- $f_m(3) = g(3, 2, 0, 1) = g(3, 3, 1, 1) = 1 + 1 = 2$
- $f_m(4) = g(4, 2, 0, 1) = g(4, 3, 1, 1) = g(4, 4, 1, 2) = 1 + 2 = 3$

- $f_m(5) = g(5, 2, 0, 1) = g(5, 3, 1, 1) = g(5, 4, 1, 2) = g(5, 5, 2, 3) = 2 + 3 = 5$
- $f_m(6) = g(6, 2, 0, 1) = g(6, 3, 1, 1) = g(6, 4, 1, 2) = g(6, 5, 2, 3)$
 $= g(6, 6, 3, 5) = 3 + 5 = 8$

En la generación de los primeros siete números de la sucesión de Fibonacci, se puede apreciar, que la suma total de llamados a las funciones f_m y g es n , para valores de $n \geq 2$; por este motivo la complejidad de la función mejorada f_m para la generación del n -ésimo término de la sucesión de Fibonacci es $\Theta(n)$.

Se presentaron dos versiones diferentes para generar el n -ésimo término de la sucesión de Fibonacci. La primera, es una recursividad “ingenua” donde se repite y se repite multiples veces el mismo proceso, lo que genera una complejidad de orden exponencial, versus una versión mejorada, donde cada elemento de la sucesión es calculado una y solo una vez, y es utilizado en la generación de los dos siguientes términos de la sucesión, siendo esta versión de orden n , donde se ha pasado de una función recursiva de complejidad exponencial a una de complejidad lineal.

Ahora surgen un par de interrogantes: ¿realmente la programación recursiva es muy costosa desde el punto de vista computacional?, o, ¿será acaso que los que afirman que la programación recursiva es muy costosa desde el punto de vista computacional sólo aprendieron a programar de forma recursiva “ingenua”?

Se invita al lector a que implemente en su lenguaje de programación favorito, las dos diferentes versiones recursivas, para calcular el n -ésimo término de la sucesión de Fibonacci, y que las ejecute para generar los elementos ubicados en las posiciones: 10, 100, 1,000, 10,000 y 100,000, para contrastar y justificar, sus resultados prácticos obtenidos (o no obtenidos) en la ejecución, versus el análisis teórico presentado en esta sección.

2.9. Ejercicios

1. Determinar el valor generado por la expresión:

$$\left\lfloor 1,3 * \left(\left\lfloor \sqrt{\left\lfloor 2,7 * \left(\left\lfloor \sqrt{\left\lfloor \sqrt{39652} \right\rfloor} \right\rfloor \right)^3} \right\rfloor} \right\rfloor \right) \right\rfloor$$

2. Producir los 17 primeros términos que se generan con las fórmulas de los siguientes items, donde Min es una función que devuelve el mínimo de dos valores, y mod es la función modulo que es equivalente al residuo de la división entera entre dos números enteros.

Explicar en palabras la forma que toma la sucesión que se genera.

- a.) $S(0) = 1$
 $S(n) = S(n-1) + (\text{Min}(2^{n \bmod 3}, 2^{(n-1) \bmod 3})) \bmod 2$, para $n \in \mathbb{Z}^+$.
- b.) $S(0) = 1$
 $S(n) = S(n-1) + (\text{Min}(2^{n \bmod 4}, 2^{(n-1) \bmod 4})) \bmod 2$, para $n \in \mathbb{Z}^+$.
- c.) $S(0) = 1$
 $S(n) = S(n-1) + (\text{Min}(2^{n \bmod 5}, 2^{(n-1) \bmod 5})) \bmod 2$, para $n \in \mathbb{Z}^+$.
- d.) $S(0) = 1$
 $S(n) = S(n-1) + (\text{Min}(3^{n \bmod 5}, 3^{(n-1) \bmod 5})) \bmod 3$, para $n \in \mathbb{Z}^+$.

3. Encontrar una fórmula para generar el n -ésimo término de cada una de las sucesión que tienen los primeros términos:

- a.) 2, 4, 6, 10, 16, 26, 42, ...
- b.) 0, 1, 1, 2, 4, 7, 13, 24, 44, 81, ...
- c.) 1, 2, 3, 10, 20, 30, 100, 200, 300, ...
- d.) $2^1, 3^1, 2^2, 3^2, 2^4, 3^4, 2^8, 3^8, 2^{16}, 3^{16}, \dots$
- e.) 0, -1, 1, -2, 2, -3, 3, -4, 4, -5, 5, ...
- f.) 1, 3, 5, 7, 9, 11, 13, 15, ...
- g.) 1, 4, 9, 16, 25, 36, 49, 64, ...
- h.) 1, 5, 14, 30, 55, 91, 140, ...
- i.) 1, 3, 4, 7, 11, 18, 29, 47, 76, ...
- j.) 1, -2, 4, -8, 16, -32, 64, -128, ...
- k.) 1, 5, 14, 30, 55, 91, ...
- l.) 1, 9, 36, 100, 225, ...
- m.) 2, 3, 5, $2^2, 3^2, 5^2, 2^3, 3^3, 5^3, 2^4, 3^4, 5^4, \dots$
- n.) 2, 5, 7, $2^2, 5^2, 7^2, 2^3, 5^3, 7^3, 2^4, 5^4, 7^4, \dots$
- ñ.) 2, 3, 5, 7, $2^2, 3^2, 5^2, 7^2, 2^3, 3^3, 5^3, 7^3, \dots$

4. Para cada uno de los siguientes items, ¿Cuál es la fórmula que representa el resultado de la suma de términos?

- a.) $1 \cdot \frac{1}{4} + 4 \cdot \frac{5}{4} + 9 \cdot \frac{9}{4} + \dots + n^2 \cdot (n - \frac{3}{4})$, donde $n \in \mathbb{Z}^+$.
- b.) $1 \cdot \frac{1}{2} + 2 \cdot \frac{3}{2} + \dots + n \cdot (n - \frac{1}{2})$, donde $n \in \mathbb{Z}^+$.
- c.) $\frac{3}{2} + \frac{14}{4} + \frac{45}{8} + \frac{124}{16} + \dots + (2n - \frac{n}{2^n})$, donde $n \in \mathbb{Z}^+$.
- d.) $1 + 4 + 7 + 10 + 13 + \dots + (3n - 2)$, donde $n \in \mathbb{Z}^+$.
- e.) $1 + 5 + 9 + 13 + 17 + \dots + (4n - 3)$, donde $n \in \mathbb{Z}^+$.
- f.) $1(2) + 2(3) + 3(4) + 4(5) + \dots + n(n+1)$, donde $n \in \mathbb{Z}^+$.
- g.) $1 \cdot \frac{2}{3} + 2 \cdot \frac{5}{3} + \dots + n \cdot (n - \frac{1}{3})$, donde $n \in \mathbb{Z}^+$.

- h.) $0 + 3 + 8 + \dots + (n^2 - 1)$, donde $n \in \mathbb{Z}^+$.
- i.) $0 + 7 + 26 + \dots + (n^3 - 1)$, donde $n \in \mathbb{Z}^+$.
- j.) $3^2 + 4^2 + 5^2 + 6^2 + \dots + (n + 2)^2?$, donde $n \in \mathbb{Z}^+$.
- k.) $1^2 + 4^2 + 7^2 + 10^2 + \dots + (3n - 2)^2?$, donde $n \in \mathbb{Z}^+$.
- l.) $4^2 + 7^2 + 10^2 + 13^2 + \dots + (3n + 1)^2?$, donde $n \in \mathbb{Z}^+$.
- m.) $5^2 + 8^2 + 11^2 + 14^2 + \dots + (3n + 2)^2?$, donde $n \in \mathbb{Z}^+$.
- n.) $5^3 + 8^3 + 11^3 + 14^3 + \dots + (3n + 2)^3?$, donde $n \in \mathbb{Z}^+$.
- ñ.) $3^3 + 4^3 + 5^3 + 6^3 + \dots + (n + 2)^3?$, donde $n \in \mathbb{Z}^+$.
- o.) $1^3 + 4^3 + 7^3 + 10^3 + \dots + (3n - 2)^3?$, donde $n \in \mathbb{Z}^+$.
- p.) $4^3 + 7^3 + 10^3 + 13^3 + \dots + (3n + 1)^3?$, donde $n \in \mathbb{Z}^+$.

5. Para la cada una de las siguientes funciones expresar el resultado en términos de n

a.) función Aloha(n : Entero Positivo)

```

r = 0
Para i = 1 Hasta n - 1
    Para j = i + 1 Hasta n
        r = r + 1
Retornar r

```

b.) función Harehare(n : Entero Positivo)

```

r = 0
Para j = 1 Hasta n
    Para k = j Hasta n
        r = r + 2k
Retornar r

```

c.) función Patumas(n : Entero Positivo)

```

r = 0
Para j = 1 Hasta n
    Para k = j Hasta 3 · j
        r = r + 3k
Retornar r

```

d.) función Conclave(n : Entero Positivo)

```

r = 0
Para j = 1 Hasta n - 2
    Para k = j + 2 Hasta n
        r = r + 2k + k2
Retornar r

```

e.) función *Camarlengo*(n : *Entero Positivo*)

$r = 0$

Para $i = 1$ *Hasta* n

Para $j = i + 1$ *Hasta* n

$r = r + j^3$

Retornar r

f.) función *Pesky*(n : *Entero Positivo*)

$r = 0$

Para $i = 1$ *Hasta* n

Para $j = i$ *Hasta* n

Para $k = j$ *Hasta* $i + j$

$r = r + 1$

Retornar r

6. Dar notación O y Ω para $f(n) = 1! + 2! + 3! + \dots + n!$. ¿Qué se puede concluir sobre la notación Θ ?

2.10. Preguntas tipo ECAES

1. Determine el valor generado por la expresión²

$$\lceil [3,3 * \lceil \sqrt{10} \rceil] + 0,01 \rceil + \lceil 0,000001 \rceil$$

- A.) 11
- B.) 12
- C.) 14
- D.) 15
- E.) 16

2. Identifique la formula que genera la secuencia de enteros: 3, 3, 3, 9, 9, 9, 9, 9, 27, 27, 27, 27, 27, 27, ...

- A.) $f(n) = 3^{\lceil \sqrt{n} \rceil}, n \geq 1$
- B.) $f(n) = 3^{\sqrt{n}}, n \geq 1$
- C.) $f(n) = 3^{\lfloor \sqrt{n} \rfloor}, n \geq 1$
- D.) $f(n) = 3^n, n \geq 1$

3. Seleccione la formula que genera la secuencia de enteros: 1, 2, 2, 2, 3, 3, 3, 3, 3, 4, 4, 4, 4, 4, 4, 4, ...

- A.) $f(n) = \lceil \sqrt{n} \rceil + 1, n \geq 0$
- B.) $f(n) = \lfloor \sqrt{n} \rfloor + 1, n \geq 1$
- C.) $f(n) = \lceil \sqrt{n+2} \rceil, n \geq 1$
- D.) $f(n) = \lfloor \sqrt{n-1} \rfloor, n \geq 1$
- E.) $f(n) = \lceil \sqrt{n} \rceil, n \geq 1$

4. Identifique la formula que genera la secuencia de enteros: 0, 1, 2, 6, 16, 44, 120, 328, 896, 2448, ...

- A.) $f(0) = 0$
 $f(1) = 1$
 $f(n) = 4f(n-1) - 2f(n-2), n \geq 2$
- B.) $f(0) = 0$
 $f(1) = 1$
 $f(n) = 2f(n-1) + 2f(n-2), n \geq 2$

²Las preguntas desde la 1 hasta la 8 son del autor.

$$\begin{aligned} \text{C.) } f(0) &= 0 \\ f(1) &= 1 \\ f(n) &= f(n-1) + 4f(n-2), \quad n \geq 2 \end{aligned}$$

$$\text{D.) } f(n) = 2^n - 1, \quad n \geq 0$$

$$\begin{aligned} \text{E.) } f(0) &= 0 \\ f(1) &= 1 \\ f(2) &= 2 \\ f(n) &= 2f(n-1) + 2f(n-2) + 2f(n-3), \quad n \geq 3 \end{aligned}$$

5. Seleccione el resultado de la sumatoria: $5^3 + 5^4 + 5^5 + \dots + 5^n$

$$\text{A.) } \frac{5^3(5^{n-2} - 1)}{4}$$

$$\text{B.) } \frac{5^3(5^{n+1} - 1)}{4}$$

$$\text{C.) } \frac{5^{n+1} - 1}{4}$$

$$\text{D.) } \frac{5^3(5^n - 1)}{4}$$

$$\text{E.) } 5^3(5^{n-2} - 1)$$

6. El resultado de la sumatoria: $3^0 - 3^1 + 3^2 - 3^3 + 3^4 - 3^5 + \dots + (-1)^n * 3^n$ es:

$$\text{A.) } \frac{(-3)^{n+1} - 1}{2}$$

$$\text{B.) } \frac{(-3)^{n+1} + 1}{4}$$

$$\text{C.) } \frac{(-3)^{n+1} + 1}{2}$$

$$\text{D.) } \frac{(-1)^n(3)^{n+1} + 1}{4}$$

$$\text{E.) } \frac{(-1)^{n+1}(3)^{n+1} + 1}{4}$$

7. Seleccione el resultado de la suma de términos: $1(2) + 2(3) + 3(4) + 4(5) + \dots + n(n+1)$

$$\text{A.) } \frac{n(n+1)(2n+1)}{3}$$

$$\text{B.) } \frac{n(n+1)(2n+1)}{6}$$

$$\text{C.) } \frac{n(n+1)(2n-1)}{3}$$

D.) $\frac{n(n+1)(n+2)}{3}$

E.) $\frac{n(n+1)(n+2)}{6}$

8. Identifique el resultado de la suma de términos: $1(3) + 2(4) + 3(5) + 4(6) + \dots + n(n+2)$

A.) $\frac{n(n+2)(2n+1)}{3}$

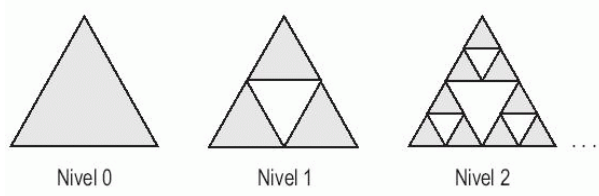
B.) $\frac{n(n+1)(2n+7)}{6}$

C.) $\frac{n(n+1)(2n-1)}{3}$

D.) $\frac{n(n+1)(n+2)}{3}$

E.) $\frac{n(n+1)(n+2)}{6}$

9. En la figura de abajo se observa una secuencia de triángulos de Sierpinski³



El proceso comienza en el Nivel 0, con un triángulo equilátero de área 1. En cada paso, a cada triángulo equilátero que queda en la figura se le elimina el triángulo formado por los segmentos de línea que unen los puntos medios de sus lados, como se ilustra en la figura. El área de la figura en el Nivel n (indicada en las figuras por el sombreado) esta dada por:

A.) $1 - \frac{1}{4^n}$

B.) $\frac{1}{4^n}$

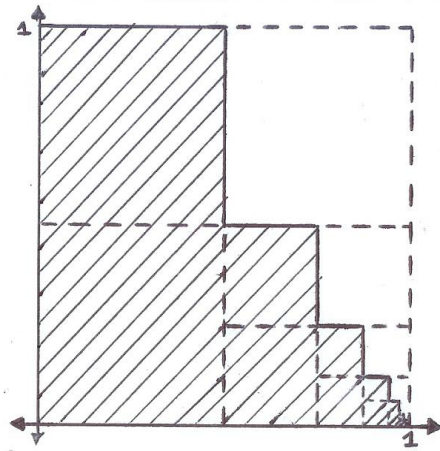
C.) $\left(\frac{3}{4}\right)^n$

D.) $1 - \left(\frac{3}{4}\right)^n$

E.) 1

³Enunciado textual del ejemplo 1 de la Guía de Orientación del Examen de Calidad de la Educación del programa de Ingeniería de Sistemas del año 2010 [MI2010]

10. La figura muestra un cuadrado de lado 1, el cual se divide en cuatro partes iguales y se somborean dos, luego se toma uno de los cuadrados de lado 0.5, se divide en cuatro partes iguales y se somborean dos, el proceso sigue así sucesivamente⁴.

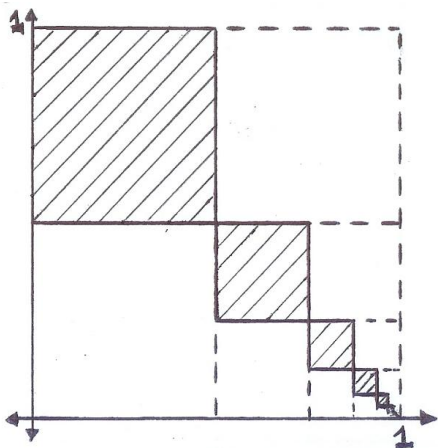


El área de la figura sombreada esta dada por:

- A.) $3/4$
 B.) $3/5$
 C.) $1/2$
 D.) $2/3$
 E.) $5/7$
11. El perímetro de la figura sombreada en el ejercicio 10 esta dado por:
- A.) 3.0
 B.) 3.25
 C.) 3.5
 D.) 3.75
 E.) 4.0
12. En la figura se observa que se tiene un cuadrado de lado 1, el cual se divide en cuatro partes iguales y se somborea una, luego se toma uno de los cuadrados de lado 0.5, se divide en cuatro partes iguales y se somborea una, el proceso sigue así sucesivamente⁵.

⁴Las preguntas 10 y 11 fueron reescritas a partir de preguntas no liberadas del ECAES de Ingeniería de Sistemas del año 2007.

⁵Las preguntas 12 y 13 son del autor.



El área de la figura sombreada esta dada por:

- A.) $1/2$
 - B.) $1/3$
 - C.) $2/3$
 - D.) $3/10$
 - E.) $4/15$
13. El perímetro de la figura sombreada en el ejercicio 12 esta dado por:
- A.) 2.0
 - B.) 2.5
 - C.) 3.0
 - D.) 3.5
 - E.) 4.0
14. Desde una altura de un metro se deja caer una pelota que rebota medio metro, luego rebota un cuarto de metro, luego rebota un octavo de metro y así sucesivamente hasta que se detiene⁶.
- ¿Cuál es la suma total en metros de la trayectoria que ha recorrido la pelota hasta que se detiene?.
- A.) 2
 - B.) 2.5
 - C.) 3
 - D.) 3.5

⁶Pregunta reescrita a partir de una pregunta no liberada del ECAES de Ingeniería de Sistemas del año 2008.

E.) 47

15. Se tienen las siguientes tres funciones definidas en los números reales positivos: $f(n) = \log(\log n)$, $g(n) = \log 2^n$ y $h(n) = \log n$. ¿Cuál de las siguientes afirmaciones es verdadera?

- A.) $f(n) = O(g(n))$ y $g(n) = O(h(n))$
 B.) $g(n) = O(h(n))$ y $h(n) = O(f(n))$
 C.) $h(n) = O(f(n))$ y $f(n) = O(g(n))$
 D.) $f(n) = O(h(n))$ y $h(n) = O(g(n))$
 E.) $h(n) = O(g(n))$ y $g(n) = O(f(n))$

16. Sea la siguiente definición recursiva, donde n es un número entero no negativo:

$$f(n) = \begin{cases} 0 & \text{si } n = 0 \\ 1 & \text{si } n = 1 \\ f(n-2) + f(n-1) & \text{si } n > 1 \end{cases}$$

¿Cuál es el resultado de $f(5)$?

- A.) 2
 B.) 3
 C.) 4
 D.) 5
 E.) 6

17. Sea la siguiente definición recursiva, donde n es un número entero no negativo:

$$f(n) = \begin{cases} 0 & \text{si } n = 0 \\ 1 & \text{si } n = 1 \\ f(n-2) + f(n-1) & \text{si } n > 1 \end{cases}$$

¿Cuál es la expresión que mejor representa la complejidad en el **mejor** de los casos de $f(n)$?, es decir, ¿cuál es la mínima cantidad de llamados que se hacen a la función f para calcular el resultado cuando ésta recibe el valor n ?

- A.) $\Omega(1)$
 B.) $\Omega(\log n)$
 C.) $\Omega(n)$

D.) $\Omega(n^2)$

E.) $\Omega(2^{\lfloor n/2 \rfloor})$

18. Sea la siguiente definición recursiva, donde n es un número entero no negativo:

$$f(n) = \begin{cases} 0 & \text{si } n = 0 \\ 1 & \text{si } n = 1 \\ f(n-2) + f(n-1) & \text{si } n > 1 \end{cases}$$

¿Cuál es la expresión que mejor representa la complejidad en el **peor** de los casos de $f(n)$?, es decir, ¿cuál es la máxima cantidad de llamados que se hacen a la función f para calcular el resultado cuando ésta recibe el valor n ?

A.) $O(1)$

B.) $O(\log n)$

C.) $O(n)$

D.) $O(n^2)$

E.) $O(2^n)$

19. Sea la siguiente definición recursiva, donde n es un número entero no negativo:

$$f(n) = \begin{cases} 0 & \text{si } n = 0 \\ 1 & \text{si } n = 1 \\ g(n, 2, 0, 1) & \text{si } n > 1 \end{cases}$$

$$g(n, i, a, b) = \begin{cases} a + b & \text{si } i = n \\ g(n, i+1, b, a+b) & \text{si } i < n \end{cases}$$

¿Cuál es el resultado de $f(5)$?

A.) 2

B.) 3

C.) 4

D.) 5

E.) 6

20. Sea la siguiente definición recursiva, donde n es un número entero no negativo:

$$f(n) = \begin{cases} 0 & \text{si } n = 0 \\ 1 & \text{si } n = 1 \\ g(n, 2, 0, 1) & \text{si } n > 1 \end{cases}$$

$$g(n, i, a, b) = \begin{cases} a + b & \text{si } i = n \\ g(n, i + 1, b, a + b) & \text{si } i < n \end{cases}$$

¿Cuál es la expresión que mejor representa la complejidad de la función g , la cual es la misma complejidad de la función $f(n)$?, es decir, ¿cuál es la expresión que mejor representa la cantidad de llamados que se hacen a la función g cuando la función f recibe un valor $n > 1$?

- A.) $\Theta(1)$
 - B.) $\Theta(\log n)$
 - C.) $\Theta(n)$
 - D.) $\Theta(n^2)$
 - E.) $\Theta(2^n)$
21. Sea la siguiente función recursiva, donde n es un número entero no negativo:

$$f(n) = \begin{cases} 3 & \text{si } n = 0 \\ -2 & \text{si } n = 1 \\ 3 \cdot f(n - 2) - 2 \cdot f(n - 1) & \text{si } n > 1 \end{cases}$$

¿Cuál es el valor de $f(4)$?

- A.) 10
 - B.) 18
 - C.) -32
 - D.) 103
 - E.) 200
22. Sea la siguiente definición recursiva, donde n es un número entero no negativo:

$$f(n) = \begin{cases} 3 & \text{si } n = 0 \\ -2 & \text{si } n = 1 \\ g(n, 2, 3, -2) & \text{si } n > 1 \end{cases}$$

$$g(n, i, a, b) = \begin{cases} 3 \cdot a - 2 \cdot b & \text{si } i = n \\ g(n, i + 1, b, 3 \cdot a - 2 \cdot b) & \text{si } i < n \end{cases}$$

¿Cuál es el resultado de $f(4)$?

- A.) 10
- B.) 18
- C.) -32
- D.) 103
- E.) 200

23. Sea el siguiente algoritmo:

```

1.  $p = 0$ ;
2.  $i = 1$ ;
3. while ( $i \leq n$ )
4.   {
5.      $i = i + 1$ ;
6.      $j = 1$ ;
7.     while ( $j \leq n$ )
8.       {
9.          $j = 2 * j$ ;
10.         $p = p + 1$ ;
11.      }
12.   }
```

¿Cuál es el valor final de p después de ejecutarse el algoritmo si $n = 5$?

- A.) 15
- B.) 16
- C.) 20
- D.) 21
- E.) 25

24. Para el algoritmo del ejercicio 23, ¿cuál es la complejidad con respecto al número de veces que se ejecuta la línea 10?

- A.) $\Theta(\log n)$
- B.) $\Theta(n^2 \log n)$
- C.) $\Theta(n \log n)$
- D.) $\Theta(n^2)$
- E.) $\Theta(2^n)$

Capítulo 3

Técnicas de demostración

3.1. Técnica de demostración directa.

La técnica de demostración directa es tradicionalmente la más utilizada en matemáticas, en ésta técnica se parte de la hipótesis (**H**) para llegar a la conclusión (**C**), $H \rightarrow C$.

Definición: El número entero n es par si existe un número entero k tal que $n = 2k$ y n es un número entero impar si existe un número entero k tal que $n = 2k + 1$.

Ejemplo 1:

Utilizando la técnica de demostración directa, demostrar que si n es un número entero impar, entonces n^2 es un número entero impar.

En la técnica de demostración directa para este ejemplo se tiene que la hipótesis es: “ n es un número entero impar” y que la conclusión es: “ n^2 es un número entero impar”.

Para demostrar que $Hipótesis \rightarrow Conclusión$, entonces se considera que la hipótesis es verdadera y se termina demostrando que la conclusión también es verdadera.

Si n es un número entero impar entonces, $n = 2k + 1$, donde $k \in \mathbb{Z}$. Por lo tanto se tiene que:

$$\begin{aligned} n^2 &= (2k + 1)^2 \\ &= 4k^2 + 4k + 1 \\ &= 2(2k^2 + 2k) + 1 \\ &= 2t + 1, \text{ donde } t = 2k^2 + 2k \text{ y } t \in \mathbb{Z} \end{aligned}$$

Por tanto como n^2 es de la forma $2t + 1$ la cual es la representación de un número entero impar, por lo tanto, se ha demostrado de forma directa que: “si n es un número entero

impar, entonces n^2 es un número entero impar” porque se partió de la hipótesis y se alcanzó la conclusión.

Ejemplo 2:

Demostrar que la suma de n enteros positivos consecutivos cualesquiera es divisible por n solo cuando n es un número entero impar positivo.

Sea m un número entero positivo ($m \in \mathbb{Z}^+$), m es el primer número de la secuencia de n números enteros consecutivos que se van a tomar, de esta forma se tiene:

$$\begin{aligned}
 & \underbrace{m + (m + 1) + (m + 2) + \cdots + (m + (n - 1))}_{n \text{ enteros positivos consecutivos}} \\
 &= \underbrace{(m + m + m + \cdots + m)}_{n \text{ veces}} + (0 + 1 + 2 + \cdots + (n - 1)) \\
 &= n(m) + (1 + 2 + \cdots + (n - 1)) \\
 &= n(m) + \frac{(n - 1)(n)}{2} \\
 &= n \left[m + \frac{n - 1}{2} \right] \\
 &= n \cdot x, \quad \text{donde } x = m + \frac{n - 1}{2}
 \end{aligned}$$

Ahora para que $n \cdot x$ sea divisible por n se necesita que x sea un número entero, esto se logra siempre y cuando el número n sea un entero impar porque al restarle el uno se obtiene un número par que al dividirlo por dos genera un número entero, la suma del entero m con el entero que es el resultado de la fracción genera como resultado un número entero. De esta forma queda demostrado que la suma de n enteros positivos consecutivos cualesquiera es divisible por n solo cuando n es un número entero impar positivo.

Definición: El número real r es número racional si existe un número entero¹ p y un número entero positivo² q , tales que $r = \frac{p}{q}$. Un número real r que no es racional es entonces un número irracional.

Ejemplo 3:

Demostrar utilizando la técnica de demostración directa que la suma de dos números racionales es un número racional.

¹Recordar que el conjunto de los números enteros es $\mathbb{Z} = \{ \dots, -3, -2, -1, 0, 1, 2, 3, \dots \}$.

²Recordar que el conjunto de los números enteros positivos es $\mathbb{Z}^+ = \{ 1, 2, 3, 4, 5, \dots \}$.

El enunciado: “la suma de dos números racionales es un número racional”, puede ser reescrito como: “Si r y s son números racionales, entonces, la suma de r y s da como resultado un número racional”.

En la técnica de demostración directa para este ejemplo se tiene que la hipótesis es: “ r y s son números racionales”, la conclusión es: “la suma de r y s da como resultado un número racional”.

Para demostrar que $Hipótesis \rightarrow Conclusión$, entonces se considera que la hipótesis es verdadera y se termina demostrando que la conclusión también es verdadera.

Si r y s son números reales racionales entonces, $r = \frac{p}{q}$ y $s = \frac{t}{u}$, donde $p, t \in \mathbb{Z}$ y $q, u \in \mathbb{Z}^+$. Por lo tanto se tiene que:

$$r + s = \frac{p}{q} + \frac{t}{u} = \frac{p \cdot u + q \cdot t}{q \cdot u} = \frac{m}{n}$$

donde $m = p \cdot u + q \cdot t$, como los números enteros positivos están contenidos en los números enteros, entonces, la multiplicación de números enteros positivos y números enteros da como resultado un número entero, también sucede lo mismo con la suma de números enteros y números enteros positivos, por este motivo $m \in \mathbb{Z}$.

Adicionalmente, $n = q \cdot u$, como la multiplicación de números enteros positivos da como resultado un número entero positivo, entonces $n \in \mathbb{Z}^+$.

Por tanto como $r + s$ es de la forma m/n la cual es la representación de un número racional, por lo tanto, se ha logrado demostrar de forma directa que: “Si r y s son números racionales, entonces, la suma de r y s da como resultado un número racional”, porque se a partir de la hipótesis se dedujo la conclusión.

3.2. Técnica de demostración indirecta

Muchas veces al intentar una demostración directa del resultado $H \rightarrow C$ (hipótesis entonces conclusión) se presentan dificultades o carencias de información tales, que se opta por establecer la validez del mismo demostrando la validez de una fórmula lógicamente equivalente con $H \rightarrow C$. En este caso se habla de una demostración indirecta. También se intenta a veces una demostración indirecta bien porque se presienten menos dificultades o bien porque las hipótesis que para el efecto se adoptan proporcionan más información que las que se utilizan en una demostración directa.

3.2.1. Técnica de demostración por contra-recíproca

La primer técnica de demostración indirecta es conocida con el nombre de técnica de demostración por contra-recíproca o también es conocida con el nombre de técnica

de demostración por contra-posición. Utiliza la equivalencia lógica:

$$(H \rightarrow C) \iff (\neg C \rightarrow \neg H)$$

y consiste en demostrar la validez de la implicación $\neg C \rightarrow \neg H$ con lo cual queda demostrado la validez de la implicación original $H \rightarrow C$ gracias a la equivalencia lógica.

Se explicará mucho mejor la técnica de demostración con el siguiente ejemplo:

Ejemplo 4:

Demostrar utilizando la técnica de demostración por contra-recíproca que si el producto de dos números enteros es par entonces uno por lo menos de los dos números enteros es par.

La representación del resultado en la forma $H \rightarrow C$ es la siguiente:

“Si $m \cdot n$ es par, entonces m es par o n es par”.

La representación del resultado en la forma $\neg C \rightarrow \neg H$ es la siguiente:

“Si m es impar y n es impar, entonces $m \cdot n$ es impar”.

Como m y n son números enteros impares entonces tienen la representación $m = 2t + 1$ y $n = 2s + 1$ para números enteros t y s , por lo tanto:

$$\begin{aligned} m \cdot n &= (2t + 1) \cdot (2s + 1) \\ &= 4ts + 2t + 2s + 1 \\ &= 2(2ts + t + s) + 1 \\ &= 2r + 1, \text{ donde } r = 2ts + t + s \end{aligned}$$

de esta forma queda demostrado que $m \cdot n$ es un número entero impar. Al demostrar la validez de $\neg C \rightarrow \neg H$ también queda demostrada la validez de $H \rightarrow C$, por lo tanto es cierto que “si el producto de dos números enteros es par entonces uno por lo menos de los dos números enteros es par.”

Ejemplo 5:

Demostrar utilizando la técnica de demostración por contra-recíproca que si $3n + 2$ es un número entero impar, entonces n es un número entero impar.

La representación del resultado en la forma $H \rightarrow C$ es la siguiente:

“Si $3n + 2$ es un número entero impar, entonces n es un número entero impar”.

La representación del resultado en la forma $\neg C \rightarrow \neg H$ es la siguiente:

“Si n es un número entero par, entonces $3n + 2$ es un número entero par”.

Como n es un número entero par entonces tienen la representación $n = 2t$ para algún número entero t , por lo tanto:

$$\begin{aligned} 3n + 2 &= 3(2t) + 2 \\ &= 2(3t) + 2 \\ &= 2(3t + 1) \\ &= 2r, \text{ donde } r = 3t + 1 \end{aligned}$$

de esta forma queda demostrado que $3n + 2$ es un número entero par. Al demostrar la validez de $\neg C \rightarrow \neg H$ también queda demostrada la validez de $H \rightarrow C$, por lo tanto es cierto que “Si $3n + 2$ es un número entero impar, entonces n es un número entero impar”.

3.2.2. Técnica de demostración por contradicción.

La segunda técnica de demostración indirecta es conocida con el nombre de técnica de demostración por contradicción o también conocida con el nombre de técnica de demostración por reducción al absurdo.

La técnica de demostración por contradicción sirve para ayudar a definir si un razonamiento es válido o no (si una conclusión se obtiene a partir de un conjunto de hipótesis).

Si un razonamiento es válido es porque siempre que las hipótesis sean verdaderas la conclusión también es verdadera, de esta forma la implicación $H \rightarrow C$ nunca tomará un valor falso y se presentará la equivalencia lógica: $(H \rightarrow C) \iff V_o$, donde se sigue teniendo una equivalencia lógica si se niegan ambos lados de la equivalencia, de esta forma se tiene el siguiente análisis que justifica la utilización de la técnica de demostración:

$$\begin{aligned} \overline{(H \rightarrow C)} &\iff \overline{V_o} \\ \overline{(H \vee C)} &\iff F_o \\ (H \wedge \overline{C}) &\iff F_o \end{aligned}$$

a partir de la equivalencia anterior está establecida la validez de la técnica de demostración, donde se supone la negación de la conclusión como otra hipótesis más del razonamiento y el objetivo es llegar a una contradicción (valor F_o) como conclusión a partir del nuevo conjunto de hipótesis.

Ejemplo 6:

Demostrar utilizando la técnica de demostración por contradicción que si $3n + 2$ es un número entero impar, entonces n es un número entero impar.

La representación del resultado en la forma $H \rightarrow C$ es la siguiente:

“Si $3n + 2$ es un número entero impar, entonces n es un número entero impar”.

La representación del resultado en la forma $H \wedge \neg C$ es la siguiente:

“ $3n + 2$ es un número entero impar **y** n es un número entero par”.

La representación del resultado en la forma $\neg C \wedge H$ es la siguiente:

“ n es un número entero par **y** $3n + 2$ es un número entero impar”.

Como n es un número entero par entonces tienen la representación $n = 2t$ para algún número entero t , por lo tanto:

$$\begin{aligned} 3n + 2 &= 3(2t) + 2 \\ &= 2(3t) + 2 \\ &= 2(3t + 1) \\ &= 2r, \text{ donde } r = 3t + 1 \end{aligned}$$

se deduce que “ $3n + 2$ es un número entero par” lo cual se contradice con la hipótesis que afirma que “ $3n + 2$ es un número entero impar”, de esta forma se obtiene el valor falso (F_o). Al llegar a una contradicción con el método de demostración por contradicción entonces queda demostrada la validez de $H \rightarrow C$, por lo tanto es cierto que “Si $3n + 2$ es un número entero impar, entonces n es un número entero impar”.

Ejemplo 7:

Determinar si el siguiente razonamiento es valido:

1. $(p \wedge q) \vee r$
 2. $r \rightarrow s$
 3. $(p \vee s) \rightarrow \bar{t}$
 4. $(q \vee s) \rightarrow u$
-
- $\therefore \bar{u} \rightarrow \bar{t}$

Ya se demostró la validez de éste razonamiento en el Capítulo 1 de Introducción a la Lógica Matemática, sección 1.2 de Reglas de Inferencia, ahora se va a utilizar la técnica de demostración por contradicción para demostrar de nuevo la validez de dicho razonamiento, para esto se tiene el nuevo conjunto de hipótesis y la nueva conclusión:

1. $(p \wedge q) \vee r$
 2. $r \rightarrow s$
 3. $(p \vee s) \rightarrow \bar{t}$
 4. $(q \vee s) \rightarrow u$
 5. $\bar{u} \rightarrow \bar{t}$
-
- $\therefore F_o$

	Pasos	Razones
1.	$(p \vee r) \wedge (q \vee r)$	Equivalencia Lógica H_1
2.	$p \vee r$	Ley de simplificación del P_1
3.	$\overline{\overline{u} \vee \bar{t}}$	Equivalencia Lógica H_5
4.	$\bar{u} \wedge t$	Equivalencia Lógica P_3
5.	t	Ley de simplificación del P_4
6.	$\bar{r} \vee s$	Equivalencia Lógica H_2
7.	$p \vee s$	Ley de resolución entre P_2 y P_6
8.	\bar{t}	Ley Modus Ponens entre P_7 e H_3
9.	$t \wedge \bar{t}$	Ley de conjunción entre P_5 y P_8
10.	F_o	Equivalencia Lógica P_9

Como se obtiene una contradicción (F_o) entonces queda demostrada la validez del razonamiento original por medio del uso de la técnica de demostración por contradicción.

Ejemplo 8:

Determinar si el siguiente razonamiento es valido:

1. $u \rightarrow r$
 2. $(r \wedge s) \rightarrow (p \vee t)$
 3. $q \rightarrow (u \wedge s)$
 4. \bar{t}
-
- $\therefore q \rightarrow p$

Ya se demostró la validez de éste razonamiento en el Capítulo 1 de Introducción a la Lógica Matemática, sección 1.2 de Reglas de Inferencia, ahora se va a utilizar la técnica de demostración por contradicción para demostrar de nuevo la validez de dicho razonamiento, para esto se tiene el nuevo conjunto de hipótesis y la nueva conclusión:

1. $u \rightarrow r$
 2. $(r \wedge s) \rightarrow (p \vee t)$
 3. $q \rightarrow (u \wedge s)$
 4. \bar{t}
 5. $\overline{q \rightarrow p}$
-
- $\therefore F_o$

	Pasos	Razones
1.	$\overline{q \vee p}$	Equivalencia Lógica H_5
2.	$q \wedge \overline{p}$	Equivalencia Lógica P_1
3.	q	Ley de Simplificación P_2
4.	\overline{p}	Ley de Simplificación P_2
5.	$\overline{p} \wedge \overline{t}$	Ley de Conjunción entre P_4 e H_4
6.	$\overline{p \vee t}$	Equivalencia Lógica P_5
7.	$u \wedge s$	Ley Modus Ponens entre P_3 e H_3
8.	$\overline{r \wedge s}$	Ley Modus Tollens entre P_6 e H_2
9.	$\overline{r} \vee \overline{s}$	Equivalencia Lógica P_8
10.	$\overline{u} \vee r$	Equivalencia Lógica H_1
11.	$\overline{u} \vee \overline{s}$	Ley de Resolución entre P_{10} y P_9
12.	$\overline{u \wedge s}$	Equivalencia Lógica P_{11}
13.	$(u \wedge s) \wedge \overline{(u \wedge s)}$	Ley de Conjunción entre P_7 y P_{12}
14.	F_o	Equivalencia Lógica P_{13}

Como se obtiene una contradicción (F_o) entonces queda demostrada la validez del razonamiento original por medio del uso de la técnica de demostración por contradicción.

3.3. Técnica de demostración por disyunción de casos

Para probar una implicación de la forma $(p_1 \vee p_2 \vee p_3 \vee \cdots \vee p_n) \rightarrow q$ se utiliza la siguiente equivalencia lógica $(p_1 \rightarrow q) \wedge (p_2 \rightarrow q) \wedge (p_3 \rightarrow q) \wedge \cdots \wedge (p_n \rightarrow q)$, la cual se obtiene de la siguiente forma:

$$(p_1 \vee p_2 \vee p_3 \vee \cdots \vee p_n) \rightarrow q$$

$$\iff \overline{(p_1 \vee p_2 \vee p_3 \vee \cdots \vee p_n)} \vee q, \quad \text{Equivalencia Lógica de la implicación.}$$

$$\iff (\overline{p_1} \wedge \overline{p_2} \wedge \overline{p_3} \wedge \cdots \wedge \overline{p_n}) \vee q, \quad \text{Ley de De Morgan.}$$

$$\iff (\overline{p_1} \vee q) \wedge (\overline{p_2} \vee q) \wedge (\overline{p_3} \vee q) \wedge \cdots \wedge (\overline{p_n} \vee q), \quad \text{Ley Distributiva.}$$

$$\iff (p_1 \rightarrow q) \wedge (p_2 \rightarrow q) \wedge (p_3 \rightarrow q) \wedge \cdots \wedge (p_n \rightarrow q), \quad \text{Eq. Lógica de la implicación.}$$

La equivalencia lógica evidencia que la implicación original con una hipótesis que se forma de la disyunción de las proposiciones $p_1, p_2, p_3, \dots, p_n$ puede ser probado al demostrar individualmente cada una de las n implicaciones $p_i \rightarrow q$, para $1 \leq i \leq n$. La única forma para que se cumpla la implicación original es cuando se cumplen absolutamente todas las implicaciones $p_1 \rightarrow q, p_2 \rightarrow q, p_3 \rightarrow q, \dots, p_n \rightarrow q$.

Ejemplo 9:

Probar o refutar utilizando el método de demostración por disyunción de casos que $n(n^2 + 5)$ es divisible por 6, para $n \in \mathbb{N}$.

Cualquier número natural n está en alguno de los siguientes seis casos, donde $k \in \mathbb{N}$:

- caso 1: $n = 6k$, el número n es un múltiplo de seis.
- caso 2: $n = 6k + 1$, el número n es un múltiplo de seis más uno.
- caso 3: $n = 6k + 2$, el número n es un múltiplo de seis más dos.
- caso 4: $n = 6k + 3$, el número n es un múltiplo de seis más tres.
- caso 5: $n = 6k + 4$, el número n es un múltiplo de seis más cuatro.
- caso 6: $n = 6k + 5$, el número n es un múltiplo de seis más cinco.

Ahora se tienen que demostrar cada uno de los seis casos, para lo cual se tiene:

caso 1: $n = 6k$:

$$\begin{aligned}
 n(n^2 + 5) &= 6k((6k)^2 + 5) \\
 &= 6k(36k^2 + 5) \\
 &= 6(36k^3 + 5k) \\
 &= 6z, \text{ donde } z = 36k^3 + 5k \text{ y } z \in \mathbb{N}.
 \end{aligned}$$

Se cumple el caso 1, porque cualquier número que es múltiplo de 6 también es divisible por 6.

caso 2: $n = 6k + 1$:

$$\begin{aligned}
 n(n^2 + 5) &= (6k + 1)((6k + 1)^2 + 5) \\
 &= (6k + 1)(36k^2 + 12k + 1 + 5) \\
 &= (6k + 1)(36k^2 + 12k + 6) \\
 &= (6k + 1) \cdot 6 \cdot (6k^2 + 2k + 1) \\
 &= 6 \cdot (6k + 1)(6k^2 + 2k + 1) \\
 &= 6z \text{ donde } z = (6k + 1)(6k^2 + 2k + 1) \text{ y } z \in \mathbb{N}.
 \end{aligned}$$

Se cumple el caso 2, porque cualquier número que es múltiplo de 6 también es divisible por 6.

caso 3: $n = 6k + 2$:

$$\begin{aligned}
 n(n^2 + 5) &= (6k + 2)((6k + 2)^2 + 5) \\
 &= (6k + 2)(36k^2 + 24k + 4 + 5) \\
 &= (6k + 2)(36k^2 + 24k + 9) \\
 &= 2 \cdot (3k + 1) \cdot 3 \cdot (12k^2 + 8k + 3) \\
 &= 6 \cdot (3k + 1)(12k^2 + 8k + 3) \\
 &= 6z, \text{ donde } z = (3k + 1)(12k^2 + 8k + 3) \text{ y } z \in \mathbb{N}.
 \end{aligned}$$

Se cumple el caso 3, porque cualquier número que es múltiplo de 6 también es divisible por 6.

caso 4: $n = 6k + 3$:

$$\begin{aligned}
 n(n^2 + 5) &= (6k + 3)((6k + 3)^2 + 5) \\
 &= (6k + 3)(36k^2 + 36k + 9 + 5) \\
 &= (6k + 3)(36k^2 + 36k + 14) \\
 &= 3 \cdot (2k + 1) \cdot 2 \cdot (18k^2 + 18k + 7) \\
 &= 6 \cdot (2k + 1)(18k^2 + 18k + 7) \\
 &= 6z, \text{ donde } z = (2k + 1)(18k^2 + 18k + 7) \text{ y } z \in \mathbb{N}.
 \end{aligned}$$

Se cumple el caso 4, porque cualquier número que es múltiplo de 6 también es divisible por 6.

caso 5: $n = 6k + 4$:

$$\begin{aligned}
 n(n^2 + 5) &= (6k + 4)((6k + 4)^2 + 5) \\
 &= (6k + 4)(36k^2 + 48k + 16 + 5) \\
 &= (6k + 4)(36k^2 + 48k + 21) \\
 &= 2 \cdot (3k + 2) \cdot 3 \cdot (12k^2 + 16k + 7) \\
 &= 6 \cdot (3k + 2)(12k^2 + 16k + 7) \\
 &= 6z, \text{ donde } z = (3k + 2)(12k^2 + 16k + 7) \text{ y } z \in \mathbb{N}.
 \end{aligned}$$

Se cumple el caso 5, porque cualquier número que es múltiplo de 6 también es divisible por 6.

caso 6: $n = 6k + 5$:

$$\begin{aligned}
 n(n^2 + 5) &= (6k + 5)((6k + 5)^2 + 5) \\
 &= (6k + 5)(36k^2 + 60k + 25 + 5) \\
 &= (6k + 5)(36k^2 + 60k + 30)
 \end{aligned}$$

$$\begin{aligned}
&= (6k + 5) \cdot 6 \cdot (6k^2 + 10k + 5) \\
&= 6 \cdot (6k + 5)(6k^2 + 10k + 5) \\
&= 6z, \text{ donde } z = (6k + 5)(6k^2 + 10k + 5) \text{ y } z \in \mathbb{N}.
\end{aligned}$$

Se cumple el caso 6, porque cualquier número que es múltiplo de 6 también es divisible por 6.

Como se cumplen todos los seis casos entonces queda demostrado utilizando la técnica de demostración por disyunción de casos que $n(n^2 + 5)$ es divisible por 6, para $n \in \mathbb{N}$.

Ejemplo 10:

Probar o refutar que el cuadrado de todo número natural, es un múltiplo de 5, ó difiere de un múltiplo de 5 en 1.

Cualquier número natural n está en alguno de los siguientes cinco casos, donde $m \in \mathbb{N}$:

- caso 1: $n = 5m$, el número n es un múltiplo de cinco.
- caso 2: $n = 5m + 1$, el número n es un múltiplo de cinco más uno.
- caso 3: $n = 5m + 2$, el número n es un múltiplo de cinco más dos.
- caso 4: $n = 5m + 3$, el número n es un múltiplo de cinco más tres.
- caso 5: $n = 5m + 4$, el número n es un múltiplo de cinco más cuatro.

Ahora se tienen que demostrar cada uno de los cinco casos, para lo cual se tiene:

- caso 1: $n = 5m$:

$$\begin{aligned}
n^2 &= (5m)^2 \\
&= 5^2 m^2 \\
&= 5(5m^2) \\
&= 5 \cdot z, \text{ donde } z = 5m^2 \text{ y } z \in \mathbb{N}.
\end{aligned}$$

Se cumple el caso 1, porque el número que se obtiene es un múltiplo de cinco.

- caso 2: $n = 5m + 1$:

$$\begin{aligned}
n^2 &= (5m + 1)^2 \\
&= 5^2 m^2 + 2 \cdot 5m + 1 \\
&= 5(5m^2 + 2m) + 1 \\
&= 5 \cdot z + 1, \text{ donde } z = 5m^2 + 2m \text{ y } z \in \mathbb{N}.
\end{aligned}$$

Se cumple el caso 2, porque el número que se obtiene es un múltiplo de cinco más uno.

- caso 3: $n = 5m + 2$:

$$\begin{aligned}
 n^2 &= (5m + 2)^2 \\
 &= 5^2m^2 + 2 \cdot 5 \cdot 2m + 4 \\
 &= 5^2m^2 + 5 \cdot 4m + 5 - 1 \\
 &= 5(5m^2 + 4m + 1) - 1 \\
 &= 5z - 1, \text{ donde } z = 5m^2 + 4m + 1 \text{ y } z \in \mathbb{N}.
 \end{aligned}$$

Se cumple el caso 3, porque el número que se obtiene es un múltiplo de cinco menos uno.

- caso 4: $n = 5m + 3$:

$$\begin{aligned}
 n^2 &= (5m + 3)^2 \\
 &= 5^2m^2 + 2 \cdot 5 \cdot 3m + 9 \\
 &= 5^2m^2 + 5 \cdot 2 \cdot 3m + 10 - 1 \\
 &= 5(5m^2 + 6m + 2) - 1 \\
 &= 5z - 1, \text{ donde } z = 5m^2 + 6m + 2 \text{ y } z \in \mathbb{N}.
 \end{aligned}$$

Se cumple el caso 4, porque el número que se obtiene es un múltiplo de cinco menos uno.

- caso 5: $n = 5m + 4$:

$$\begin{aligned}
 n^2 &= (5m + 4)^2 \\
 &= 5^2m^2 + 2 \cdot 5 \cdot 4m + 16 \\
 &= 5^2m^2 + 5 \cdot 2 \cdot 4m + 15 + 1 \\
 &= 5(5m^2 + 8m + 3) + 1 \\
 &= 5z + 1, \text{ donde } z = 5m^2 + 8m + 3 \text{ y } z \in \mathbb{N}.
 \end{aligned}$$

Se cumple el caso 5, porque el número que se obtiene es un múltiplo de cinco más uno.

Como se cumplen todos los cinco casos entonces queda demostrado utilizando la técnica de demostración por disyunción de casos que el cuadrado de todo número natural, es un múltiplo de 5, ó difiere de un múltiplo de 5 en 1.

Ejemplo 11:

Probar o refutar que $n^2 - 1$ es divisible por 8 para los números enteros impares.

Cualquier número entero n está en alguno de los siguientes ocho casos, donde $k \in \mathbb{Z}$:

- caso 1: $n = 8k$, el número n es un múltiplo de ocho.
- caso 2: $n = 8k + 1$, el número n es un múltiplo de ocho más uno.
- caso 3: $n = 8k + 2$, el número n es un múltiplo de ocho más dos.
- caso 4: $n = 8k + 3$, el número n es un múltiplo de ocho más tres.
- caso 5: $n = 8k + 4$, el número n es un múltiplo de ocho más cuatro.
- caso 6: $n = 8k + 5$, el número n es un múltiplo de ocho más cinco.
- caso 7: $n = 8k + 6$, el número n es un múltiplo de ocho más seis.
- caso 8: $n = 8k + 7$, el número n es un múltiplo de ocho más siete.

En el análisis sólo se consideraran los casos 2, 4, 6 y 8 porque estos son los que representan a los números enteros impares. Para la demostración de estos cuatro casos se tiene:

caso 2: $n = 8k + 1$:

$$\begin{aligned}
 n^2 - 1 &= (8k + 1)^2 - 1 \\
 &= 64k^2 + 16k + 1 - 1 \\
 &= 8(8k^2 + 2k) \\
 &= 8z, \text{ donde } z = 8k^2 + 2k \text{ y } z \in \mathbb{N}
 \end{aligned}$$

Se cumple el caso 2, porque cualquier número que es múltiplo de 8 también es divisible por 8.

caso 4: $n = 8k + 3$:

$$\begin{aligned}
 n^2 - 1 &= (8k + 3)^2 - 1 \\
 &= 64k^2 + 48k + 9 - 1 \\
 &= 64k^2 + 48k + 8 \\
 &= 8(8k^2 + 6k + 1) \\
 &= 8z, \text{ donde } z = 8k^2 + 6k + 1 \text{ y } z \in \mathbb{N}
 \end{aligned}$$

Se cumple el caso 4, porque cualquier número que es múltiplo de 8 también es divisible por 8.

caso 6: $n = 8k + 5$:

$$\begin{aligned}
 n^2 - 1 &= (8k + 5)^2 - 1 \\
 &= 64k^2 + 80k + 25 - 1 \\
 &= 64k^2 + 80k + 24
 \end{aligned}$$

$$\begin{aligned}
&= 8(8k^2 + 10k + 3) \\
&= 8z, \text{ donde } z = 8k^2 + 10k + 3 \text{ y } z \in \mathbb{N}
\end{aligned}$$

Se cumple el caso 6, porque cualquier número que es múltiplo de 8 también es divisible por 8.

caso 8: $n = 8k + 7$:

$$\begin{aligned}
n^2 - 1 &= (8k + 7)^2 - 1 \\
&= 64k^2 + 8(14)k + 49 - 1 \\
&= 64k^2 + 8(14)k + 48 \\
&= 8(8k^2 + 14k + 6) \\
&= 8z, \text{ donde } z = 8k^2 + 14k + 6 \text{ y } z \in \mathbb{N}
\end{aligned}$$

Se cumple el caso 8, porque cualquier número que es múltiplo de 8 también es divisible por 8.

Como se cumplen todos los cuatro casos que representan números enteros impares, entonces queda demostrado utilizando la técnica de demostración por disyunción de casos que $n^2 - 1$ es divisible por 8 para todo n que es un número entero impar.

3.4. Técnica de demostración por contraejemplo

La técnica de demostración por contraejemplo es utilizada para demostrar que un argumento que se intuye que no es válido realmente no es válido. La técnica de demostración por contraejemplo no sirve para demostrar validez, únicamente sirve para demostrar falsedad, por este motivo el hecho de no encontrar un contraejemplo no garantiza la validez del argumento, en dicho caso se debe utilizar alguna de las técnicas de demostración de éste capítulo que sirva para demostrar validez.

Ejemplo 12:

Demostrar o refutar si el siguiente razonamiento es válido:

1. $p \rightarrow (q \rightarrow r)$
 2. $q \rightarrow (p \rightarrow r)$
-
- $\therefore (p \vee q) \rightarrow r$

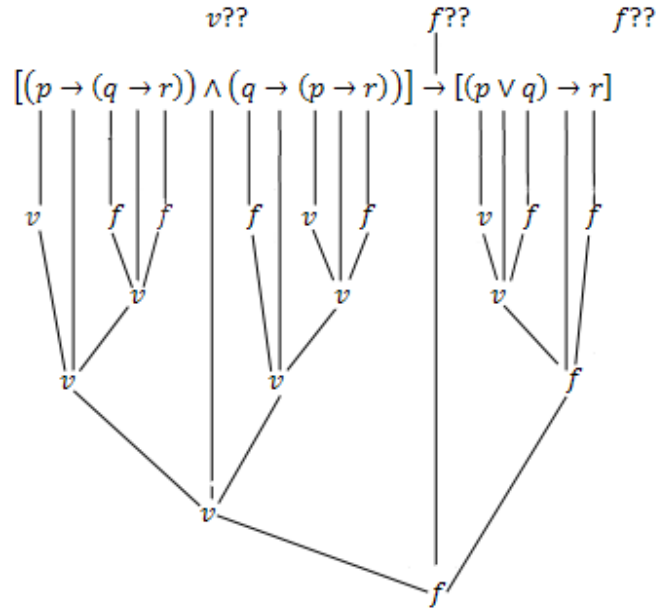
Para este ejemplo primero que todo se va a tratar de determinar la validez del razonamiento por medio de la técnica de demostración directa al hacer uso de las dos hipótesis, de equivalencias lógicas y de reglas de inferencia para llegar a la conclusión, de esta forma se tiene:

Pasos	Razones
1. $\bar{p} \vee (\bar{q} \vee r)$	Equivalencia Lógica de H_1
2. $\bar{q} \vee (\bar{p} \vee r)$	Equivalencia Lógica del P_1
3. $q \rightarrow (p \rightarrow r)$	Equivalencia Lógica del P_2
4. $(q \rightarrow (p \rightarrow r)) \wedge (q \rightarrow (p \rightarrow r))$	Ley de conjunción entre P_3 e H_2
5. $q \rightarrow (p \rightarrow r)$	Equivalencia Lógica del P_4

No se llegó a la conclusión, esto indica que “posiblemente” el razonamiento no es correcto, como se intuye que el razonamiento no es válido entonces se va a buscar una asignación de valores de verdad para las variables proposicionales que hagan que todas las hipótesis sean verdaderas y la conclusión sea falsa, para esto se debe tener en cuenta que el razonamiento puede ser representado de forma equivalente por la expresión:

$$[(p \rightarrow (q \rightarrow r)) \wedge (q \rightarrow (p \rightarrow r))] \rightarrow [(p \vee q) \rightarrow r]$$

donde si todas las hipótesis son verdaderas entonces el antecedente de la implicación sería verdadero y si la conclusión es falsa entonces el consecuente de la implicación sería falso los que llevaría a que la implicación fuera falsa, o dicho de forma equivalente, lo que haría que el razonamiento sea falso. En la exploración de la posible asignación de valores de verdad para las variables proposicionales que hagan que el razonamiento sea falso se tiene:



en consecuencia, la asignación de los valores $p: V_o, q: F_o, r: F_o$, hacen que el razonamiento sea falso, por lo tanto dicha asignación de valores son un contraejemplo de la validez del razonamiento.

Ejemplo 13:

¿Es posible probar que $\binom{2n}{n} = 2\binom{n}{2} + n^2$, para $n \in \mathbb{Z}^+$ donde $n \geq 2$?

Como dato anecdótico al autor, en un examen de Matemáticas Discretas en la Maestría, se le pidió que demostrara dicho ejercicio, cuando se pide que se demuestre se sobre entiende que es cierta la fórmula, por éste motivo durante aproximadamente tres horas trato de multiples formas de demostrar la validez de dicha fórmula, pero no lo logro. Existía la posibilidad de que le hubieran pedido que demostrara algo que no se podía demostrar, por este motivo utilizó la técnica de demostración por contraejemplo para demostrar que la fórmula era falsa, lo que se hizo fue evaluar la fórmula a partir de números enteros positivos mayores o iguales a dos para ver si se cumple o no, para esto se tiene:

Evaluación de la fórmula en $n = 2$:

$$\begin{aligned}\binom{2(2)}{2} &= 2\binom{2}{2} + 2^2 \\ \binom{4}{2} &= 2\binom{2}{2} + 2^2 \\ \frac{4!}{(4-2)! \cdot 2!} &= 2\left[\frac{2!}{(2-2)! \cdot 2!}\right] + 4 \\ \frac{4!}{2! \cdot 2!} &= 2\left[\frac{2!}{0! \cdot 2!}\right] + 4 \\ \frac{4 \cdot 3 \cdot 2!}{2! \cdot 2!} &= 2\left[\frac{2!}{0! \cdot 2!}\right] + 4 \\ \frac{4 \cdot 3}{2!} &= 2\left[\frac{1}{0!}\right] + 4 \\ \frac{4 \cdot 3}{2} &= 2\left[\frac{1}{1}\right] + 4 \\ 6 &= 2 + 4 \\ 6 &= 6\end{aligned}$$

La fórmula se cumple para $n = 2$

Evaluación de la fórmula en $n = 3$:

$$\begin{aligned}\binom{2(3)}{3} &= 2\binom{3}{2} + 3^2 \\ \binom{6}{3} &= 2\binom{3}{2} + 3^2\end{aligned}$$

$$\frac{6!}{(6-3)! \cdot 3!} = 2 \left[\frac{3!}{(3-2)! \cdot 2!} \right] + 9$$

$$\frac{6!}{3! \cdot 3!} = 2 \left[\frac{3!}{1! \cdot 2!} \right] + 9$$

$$\frac{6 \cdot 5 \cdot 4 \cdot 3!}{3! \cdot 3!} = 2 \left[\frac{3 \cdot 2!}{1! \cdot 2!} \right] + 9$$

$$\frac{6 \cdot 5 \cdot 4}{3!} = 2 \left[\frac{3}{1!} \right] + 9$$

$$\frac{6 \cdot 5 \cdot 4}{6} = 2 \left[\frac{3}{1} \right] + 9$$

$$\frac{5 \cdot 4}{1} = 2[3] + 9$$

$$20 = 6 + 9$$

$$20 = 15$$

La fórmula no se cumple para $n = 3$.

Como se encontraron un valor de n para el cual no se cumple la fórmula, entonces queda demostrado por contra ejemplo que la fórmula es falsa.

3.5. Técnica de demostración por inducción matemática

La técnica de demostración por inducción matemática es utilizada para probar proposiciones de la forma $\forall_n p(n)$, donde el universo del discurso es el conjunto de los números naturales (\mathbb{N}).

Una demostración por la técnica de inducción matemática consiste de tres pasos:

Paso base:

se demuestra la validez de la proposición p evaluada en el caso base, donde dicho caso base puede ser un cero o un uno dependiendo del punto de partida o condición inicial del problema que se está demostrando.

Paso inductivo (o hipótesis de inducción):

se asume que es verdadera la proposición p evaluada en un número natural k .

Paso post-inductivo:

apoyados en la suposición de validez de la proposición $p(k)$ se demuestra la validez de la proposición $p(k+1)$, es decir, $p(k) \rightarrow p(k+1)$.

Cuando se cumplen los tres casos de la técnica por inducción matemática, entonces se ha demostrado que la proposición $p(n)$ es verdadero para todo número natural n , es decir, se ha demostrado que $\forall_n p(n)$ es verdadero.

Ejemplo 14:

Probar o refutar utilizando la técnica de demostración por inducción matemática que:

$$\sum_{i=1}^n i^2 = 1^2 + 2^2 + 3^2 + \dots + n^2 = \frac{n(n+1)(2n+1)}{6}, \text{ para } n \in \mathbb{Z}^+$$

Para el desarrollo de la demostración considerar que se tiene la proposición

$$p(n) = \sum_{i=1}^n i^2 = 1^2 + 2^2 + 3^2 + \dots + n^2 = \frac{n(n+1)(2n+1)}{6}, \text{ donde } n \in \mathbb{Z}^+$$

La proposición $p(n)$ tomará el valor verdadero (V_o) o falso (F_o) dependiendo de si se cumple o no la igualdad $\sum_{i=1}^n i^2 = \frac{n(n+1)(2n+1)}{6}$. Ahora se consideran los tres pasos de la técnica por inducción matemática:

Paso base $n = 1$:

$$p(1) = \underbrace{\sum_{i=1}^1 i^2 = 1^2 = 1}_{\text{resultado a partir de la sumatoria de términos}} = \underbrace{\frac{1(1+1)(2(1)+1)}{6} = \frac{1(2)(3)}{6}}_{\text{resultado a partir de la fórmula}} = 1$$

como se obtiene el mismo resultado en la sumatoria de términos y en la fórmula que es la solución de la sumatoria entonces la proposición $p(1)$ es verdadera y la demostración continua en el paso inductivo.

Paso inductivo $n = k$:

$$p(k) = \sum_{i=1}^k i^2 = 1^2 + 2^2 + 3^2 + \dots + k^2 = \frac{k(k+1)(2k+1)}{6}, \text{ se asume que la proposición } p(k) \text{ es verdadera, esto quiere decir, que se asume que se cumple la siguiente igualdad } 1^2 + 2^2 + 3^2 + \dots + k^2 = \frac{k(k+1)(2k+1)}{6}.$$

Paso post-inductivo $n = k + 1$:

$$p(k+1) = \sum_{i=1}^{k+1} i^2 = \underbrace{1^2 + 2^2 + 3^2 + \dots + k^2}_{\substack{\text{Se reemplaza por su equivalente} \\ \text{en el paso inductivo}}} + (k+1)^2 = \frac{(k+1)(k+1+1)(2(k+1)+1)}{6}$$

$$\frac{k(k+1)(2k+1)}{6} + (k+1)^2 = \frac{(k+1)(k+2)(2k+2+1)}{6}$$

$$\begin{aligned}
(k+1) \left[\frac{k(2k+1)}{6} + (k+1) \right] &= \frac{(k+1)(k+2)(2k+3)}{6} \\
(k+1) \left[\frac{k(2k+1) + 6(k+1)}{6} \right] &= \frac{(k+1)(k+2)(2k+3)}{6} \\
(k+1) \left[\frac{2k^2 + k + 6k + 6}{6} \right] &= \frac{(k+1)(k+2)(2k+3)}{6} \\
(k+1) \left[\frac{2k^2 + 7k + 6}{6} \right] &= \frac{(k+1)(k+2)(2k+3)}{6}
\end{aligned}$$

hace falta averiguar a que es igual la ecuación cuadrática $2k^2 + 7k + 6 = 0$, para esto se debe recordar primero la fórmula general para este fin, la cual es:

$$k = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

en la ecuación cuadrática que estamos trabajando se tiene que $a = 2$, $b = 7$ y $c = 6$, al reemplazar en la fórmula se tiene:

$$k = \frac{-7 \pm \sqrt{7^2 - (4)(2)(6)}}{2 \cdot 2} = \frac{-7 \pm \sqrt{49 - 48}}{4} = \frac{-7 \pm \sqrt{1}}{4} = \frac{-7 \pm 1}{4}$$

de donde se obtienen las raíces reales distintas

$$\begin{aligned}
\blacksquare k &= \frac{-7 - 1}{4} = -\frac{8}{4} = -2, & k &= -2, & k+2 &= 0 \\
\blacksquare k &= \frac{-7 + 1}{4} = -\frac{6}{4} = -\frac{3}{2}, & k &= -\frac{3}{2}, & 2k+3 &= 0
\end{aligned}$$

a partir de las soluciones de la ecuación cuadrática se tiene que

$$2k^2 + 7k + 6 = (k+2)(2k+3)$$

ahora continuando con la demostración del paso post-inductivo se tiene:

$$\begin{aligned}
(k+1) \left[\frac{(k+2)(2k+3)}{6} \right] &= \frac{(k+1)(k+2)(2k+3)}{6} \\
\frac{(k+1)(k+2)(2k+3)}{6} &= \frac{(k+1)(k+2)(2k+3)}{6}
\end{aligned}$$

Se cumple la igualdad, por lo tanto la proposición $p(k+1)$ es verdadera, como se cumplen los tres pasos de técnica de demostración por inducción matemática entonces queda demostrada la validez de la solución de la sumatoria originalmente planteada.

Ejemplo 15:

Demostrar utilizando la técnica de inducción matemática que:

$$\sum_{i=1}^n i^3 = 1^3 + 2^3 + 3^3 + \dots + n^3 = \left(\frac{n(n+1)}{2} \right)^2, \text{ para } n \in \mathbb{Z}^+$$

Para el desarrollo de la demostración considerar que se tiene la proposición

$$p(n) = \sum_{i=1}^n i^3 = 1^3 + 2^3 + 3^3 + \dots + n^3 = \left(\frac{n(n+1)}{2} \right)^2, \text{ para } n \in \mathbb{Z}^+$$

la proposición $p(n)$ tomará el valor verdadero (V_o) o falso (F_o) dependiendo de si se cumple o no la igualdad $\sum_{i=1}^n i^3 = \left(\frac{n(n+1)}{2} \right)^2$. Ahora se consideran los tres pasos de la técnica por inducción matemática:

Paso base $n = 1$:

$$p(1) = \underbrace{\sum_{i=1}^1 i^3 = 1^3 = 1}_{\text{resultado a partir de la sumatoria de términos}} = \underbrace{\left(\frac{1(1+1)}{2} \right)^2 = \left(\frac{1(2)}{2} \right)^2 = 1^2 = 1}_{\text{resultado a partir de la fórmula}}$$

como se obtiene el mismo resultado en la sumatoria de términos y en la fórmula que es la solución de la sumatoria entonces la proposición $p(1)$ es verdadera y la demostración continua en el paso inductivo.

Paso inductivo $n = k$:

$$p(k) = \sum_{i=1}^k i^3 = 1^3 + 2^3 + 3^3 + \dots + k^3 = \left(\frac{k(k+1)}{2} \right)^2, \text{ se asume que la proposición } p(k) \text{ es verdadera, esto quiere decir, que se asume que se cumple la siguiente igualdad } 1^3 + 2^3 + 3^3 + \dots + k^3 = \left(\frac{k(k+1)}{2} \right)^2.$$

Paso post-inductivo $n = k + 1$:

$$p(k+1) = \sum_{i=1}^{k+1} i^3 = \underbrace{1^3 + 2^3 + 3^3 + \dots + k^3}_{\substack{\text{Se reemplaza por su equivalente} \\ \text{en el paso inductivo}}} + (k+1)^3 = \left(\frac{(k+1)(k+1+1)}{2} \right)^2$$

$$\begin{aligned} \left(\frac{k(k+1)}{2} \right)^2 + (k+1)^3 &= \left(\frac{(k+1)(k+2)}{2} \right)^2 \\ \frac{k^2(k+1)^2}{4} + (k+1)^3 &= \left(\frac{(k+1)(k+2)}{2} \right)^2 \\ (k+1)^2 \left[\frac{k^2}{4} + \frac{4(k+1)}{4} \right] &= \left(\frac{(k+1)(k+2)}{2} \right)^2 \end{aligned}$$

$$\begin{aligned}
(k+1)^2 \left[\frac{k^2 + 4k + 4}{4} \right] &= \left(\frac{(k+1)(k+2)}{2} \right)^2 \\
(k+1)^2 \left[\frac{(k+2)^2}{2^2} \right] &= \left(\frac{(k+1)(k+2)}{2} \right)^2 \\
\frac{(k+1)^2(k+2)^2}{2^2} &= \left(\frac{(k+1)(k+2)}{2} \right)^2 \\
\left(\frac{(k+1)(k+2)}{2} \right)^2 &= \left(\frac{(k+1)(k+2)}{2} \right)^2
\end{aligned}$$

Se cumple la igualdad, por lo tanto la proposición $p(k+1)$ es verdadera, como se cumplen los tres pasos de técnica de demostración por inducción matemática entonces queda demostrada la validez de la solución de la sumatoria originalmente planteada.

Ejemplo 16:

Demostrar utilizando la técnica de inducción matemática que:

$$\sum_{i=1}^n \frac{i}{2^i} = \frac{1}{2^1} + \frac{2}{2^2} + \frac{3}{2^3} + \dots + \frac{n}{2^n} = 2 - \frac{n+2}{2^n}, \text{ para } n \in \mathbb{Z}^+$$

Para el desarrollo de la demostración considerar que se tiene la proposición

$$p(n) = \sum_{i=1}^n \frac{i}{2^i} = \frac{1}{2^1} + \frac{2}{2^2} + \frac{3}{2^3} + \dots + \frac{n}{2^n} = 2 - \frac{n+2}{2^n}, \text{ para } n \in \mathbb{Z}^+$$

la proposición $p(n)$ tomará el valor verdadero (V_o) o falso (F_o) dependiendo de si se cumple o no la igualdad $\sum_{i=1}^n \frac{i}{2^i} = 2 - \frac{n+2}{2^n}$. Ahora se consideran los tres pasos de la técnica por inducción matemática:

Paso base $n = 1$:

$$\begin{aligned}
p(1) &= \underbrace{\sum_{i=1}^1 \frac{i}{2^i} = \frac{1}{2^1} = \frac{1}{2}}_{\text{resultado a partir de la sumatoria de términos}} = \underbrace{2 - \frac{1+2}{2^1} = 2 - \frac{3}{2} = \frac{4-3}{2} = \frac{1}{2}}_{\text{resultado a partir de la fórmula}}
\end{aligned}$$

como se obtiene el mismo resultado en la sumatoria de términos y en la fórmula que es la solución de la sumatoria entonces la proposición $p(1)$ es verdadera y la demostración continua en el paso inductivo.

Paso inductivo $n = k$:

$$\begin{aligned}
p(k) &= \sum_{i=1}^k \frac{i}{2^i} = \frac{1}{2^1} + \frac{2}{2^2} + \frac{3}{2^3} + \dots + \frac{k}{2^k} = 2 - \frac{k+2}{2^k}, \text{ se asume que la proposición } p(k) \\
&\text{es verdadera, esto quiere decir, que se supone que se cumple la siguiente igualdad} \\
&\frac{1}{2^1} + \frac{2}{2^2} + \frac{3}{2^3} + \dots + \frac{k}{2^k} = 2 - \frac{k+2}{2^k}.
\end{aligned}$$

Paso post-inductivo $n = k + 1$:

$$p(k+1) = \sum_{i=1}^{k+1} \frac{i}{2^i} = \underbrace{\frac{1}{2} + \frac{2}{2^2} + \frac{3}{2^3} + \dots + \frac{k}{2^k}}_{\text{Se reemplaza por su equivalente en el paso inductivo}} + \frac{k+1}{2^{k+1}} = 2 - \frac{k+1+2}{2^{k+1}}$$

*Se reemplaza por su equivalente
en el paso inductivo*

$$\begin{aligned} 2 - \frac{k+2}{2^k} + \frac{k+1}{2^{k+1}} &= 2 - \frac{k+3}{2^{k+1}} \\ 2 - \left(\frac{k+2}{2^k} \cdot \frac{2}{2} \right) + \frac{k+1}{2^{k+1}} &= 2 - \frac{k+3}{2^{k+1}} \\ 2 - \frac{2k+4}{2^{k+1}} + \frac{k+1}{2^{k+1}} &= 2 - \frac{k+3}{2^{k+1}} \\ 2 - \left[\frac{2k+4}{2^{k+1}} - \frac{k+1}{2^{k+1}} \right] &= 2 - \frac{k+3}{2^{k+1}} \\ 2 - \left[\frac{2k+4 - (k+1)}{2^{k+1}} \right] &= 2 - \frac{k+3}{2^{k+1}} \\ 2 - \left[\frac{2k+4 - k - 1}{2^{k+1}} \right] &= 2 - \frac{k+3}{2^{k+1}} \\ 2 - \left[\frac{k+3}{2^{k+1}} \right] &= 2 - \frac{k+3}{2^{k+1}} \\ 2 - \frac{k+3}{2^{k+1}} &= 2 - \frac{k+3}{2^{k+1}} \end{aligned}$$

Se cumple la igualdad, por lo tanto la proposición $p(k+1)$ es verdadera, como se cumplen los tres pasos de técnica de demostración por inducción matemática entonces queda demostrada la validez de la solución de la sumatoria originalmente planteada.

Ejemplo 17:

En el ejemplo 24 del capítulo 2 del Uso de Sucesiones y Sumatorias en Computación, se obtuvo la siguiente solución para la sumatoria de términos:

$$\sum_{i=1}^n i \cdot \left(i^2 - \frac{4}{5} \right) = 1 \cdot \frac{1}{5} + 2 \cdot \frac{16}{5} + 3 \cdot \frac{41}{5} + \dots + n \cdot \left(n^2 - \frac{4}{5} \right) = \frac{n(n+1)(5n(n+1) - 8)}{20}$$

demostrar utilizando la técnica de inducción matemática que dicho resultado es correcto.

Para el desarrollo de la demostración considerar que se tiene la proposición:

$$p(n) = \sum_{i=1}^n i \cdot \left(i^2 - \frac{4}{5}\right) = 1 \cdot \frac{4}{5} + 2 \cdot \frac{16}{5} + 3 \cdot \frac{41}{5} + \dots + n \cdot \left(n^2 - \frac{4}{5}\right) = \frac{n(n+1)(5n(n+1) - 8)}{20}$$

Ahora se consideran los tres pasos de la técnica por inducción matemática:

Paso base n = 1:

$$\begin{aligned} p(1) &= \underbrace{\sum_{i=1}^1 i \cdot \left(i^2 - \frac{4}{5}\right)}_{\text{Resultado a partir de la sumatoria de términos}} = 1 \cdot \left(1^2 - \frac{4}{5}\right) = 1 \cdot \left(\frac{1}{5}\right) = \frac{1}{5} \\ &= \underbrace{\frac{1 \cdot (1+1) \cdot (5 \cdot 1 \cdot (1+1) - 8)}{20}}_{\text{Resultado a partir de la fórmula}} = \frac{1 \cdot (2) \cdot (5 \cdot (2) - 8)}{20} = \frac{(2)(2)}{20} = \frac{4}{20} = \frac{1}{5} \end{aligned}$$

Como se obtiene el mismo resultado en la sumatoria de términos y en la fórmula que es la solución de la sumatoria entonces la proposición $p(1)$ es verdadera y la demostración continua en el paso inductivo.

Paso inductivo n = k:

$$\begin{aligned} p(k) &= \sum_{i=1}^k i \cdot \left(i^2 - \frac{4}{5}\right) = 1 \cdot \frac{1}{5} + 2 \cdot \frac{16}{5} + 3 \cdot \frac{41}{5} + \dots + k \cdot \left(k^2 - \frac{4}{5}\right) \\ &= \frac{k(k+1)(5k(k+1) - 8)}{20} \end{aligned}$$

Se asume que la proposición $p(k)$ es verdadera.

Paso post-inductivo n = k + 1:

$$\begin{aligned} p(k+1) &= \sum_{i=1}^{k+1} i \cdot \left(i^2 - \frac{4}{5}\right) \\ &= \underbrace{1 \cdot \frac{1}{5} + 2 \cdot \frac{16}{5} + 3 \cdot \frac{41}{5} + \dots + k \cdot \left(k^2 - \frac{4}{5}\right)}_{\text{se reemplaza por su equivalente en el paso inductivo}} + (k+1) \cdot \left((k+1)^2 - \frac{4}{5}\right) \\ &= \frac{k(k+1)(5k(k+1) - 8)}{20} + (k+1) \cdot \left((k+1)^2 - \frac{4}{5}\right) \\ &= \frac{(k+1)}{20} \cdot \left[k \cdot (5k \cdot (k+1) - 8) + 20 \cdot \left((k+1)^2 - \frac{4}{5}\right) \right] \\ &= \frac{(k+1)}{20} \cdot \left[k \cdot (5k^2 + 5k - 8) + 20 \cdot (k^2 + 2k + 1) - 16 \right] \\ &= \frac{(k+1)}{20} \cdot \left[5k^3 + 5k^2 - 8k + 20k^2 + 40k + 20 - 16 \right] \\ &= \frac{(k+1)}{20} \cdot \left[5k^3 + 25k^2 + 32k + 4 \right]^3 \end{aligned}$$

$$\begin{aligned}
&= \frac{(k+1)}{20} \cdot \left[(k+2)(5k^2 + 15k + 2) \right] \\
&= \frac{(k+1)(k+2)(5k^2 + 15k + 2)}{20} \\
&= \frac{(k+1)(k+2)(5k^2 + 15k + 10 - 8)}{20} \\
&= \frac{(k+1)(k+2)(5(k^2 + 3k + 2) - 8)}{20} \\
&= \frac{(k+1)(k+2)(5(k+1)(k+2) - 8)}{20} \\
&= \frac{(k+1)((k+1)+1)(5(k+1)((k+1)+1) - 8)}{20}
\end{aligned}$$

El resultado anterior es el equivalente a reemplazar en la fórmula que se está demostrando el valor de n por $k+1$. De esta manera la proposición $p(k+1)$ toma valor verdadero. Al cumplirse los tres pasos de la técnica de demostración por inducción matemática, entonces queda demostrada la validez de la solución para la sumatoria de términos.

Ejemplo 18:

Demostrar por la técnica de inducción matemática que $7^n - 2^n$ es múltiplo de 5, para $n \in \mathbb{N}$.

Para el desarrollo de la demostración considerar que se tiene la proposición $p(n) = 7^n - 2^n = 5 \cdot p$, para $p \in \mathbb{N}$. La proposición $p(n)$ tomará el valor verdadero (V_o) o falso (F_o) dependiendo de si se cumple o no que $7^n - 2^n$ es múltiplo de 5. Ahora se consideran los tres pasos de la técnica por inducción matemática:

Paso base $n = 0$:

$$p(0) = 7^0 - 2^0 = 1 - 1 = 0 = 5(0)$$

la proposición $p(0)$ es verdadera porque al evaluar $7^0 - 2^0$ se obtiene como resultado el número 0 el cual es múltiplo de 5.

Paso inductivo $n = k$:

$p(k) = 7^k - 2^k = 5 \cdot m$, para $m \in \mathbb{Z}^+$. Se asume que la proposición $p(k)$ es

$$^3\text{Haciendo uso de la división sintética: } -2 \left| \begin{array}{rrrr} 5 & 25 & 32 & 4 \\ & -10 & -30 & -4 \\ \hline 5 & 15 & 2 & 0 \end{array} \right.$$

se obtiene que $5k^3 + 25k^2 + 32k + 4 = (k+2)(5k^2 + 15k + 2)$.

$$\begin{array}{cccccccccccccccc}
& & & & & & & & 1 & & & & & & & & & \\
& & & & & & & & 1 & & 1 & & & & & & & \\
& & & & & & & 1 & & 2 & & 1 & & & & & & \\
& & & & & & 1 & & 3 & & 3 & & 1 & & & & & \\
& & & & 1 & & 4 & & 6 & & 4 & & 1 & & & & & \\
& & 1 & & 5 & & 10 & & 10 & & 5 & & 1 & & & & & \\
1 & & 6 & & 15 & & 20 & & 15 & & 6 & & 1 & & & & & \\
\vdots & & & & & & \vdots & & & & & & \vdots & & & & &
\end{array}$$

En el Triángulo de Pascal se evidencian las siguientes reglas de generación:

- Únicamente el número uno está en la cúspide y en los lados del triángulo.
- Los números internos del triángulo se obtienen al sumar los dos números más cercanos a este en el nivel inmediatamente superior.

El Triángulo de Pascal también se puede generar utilizando el combinatorio de la siguiente forma:

$$\begin{array}{cccccccc}
 & & & & \binom{0}{0} & & & \\
 & & & & \binom{1}{0} & & \binom{1}{1} & \\
 & & & \binom{2}{0} & & \binom{2}{1} & & \binom{2}{2} \\
 & & \binom{3}{0} & & \binom{3}{1} & & \binom{3}{2} & & \binom{3}{3} \\
 & \binom{4}{0} & & \binom{4}{1} & & \binom{4}{2} & & \binom{4}{3} & & \binom{4}{4} \\
 \binom{5}{0} & & \binom{5}{1} & & \binom{5}{2} & & \binom{5}{3} & & \binom{5}{4} & & \binom{5}{5} \\
 \binom{6}{0} & & \binom{6}{1} & & \binom{6}{2} & & \binom{6}{3} & & \binom{6}{4} & & \binom{6}{5} & & \binom{6}{6} \\
 \vdots & & & & \vdots & & & & \vdots & & & &
 \end{array}$$

Al “cruzar” las dos formas de generar el Triángulo de Pascal se obtiene la siguiente definición recursiva que sirve para calcular n combinado r , donde $n, r \in \mathbb{N}$:

$$\binom{n}{r} = \begin{cases} 1 & \text{si } r = 0 \\ 1 & \text{si } n = r \\ \binom{n-1}{r-1} + \binom{n-1}{r} & \text{si } n > r \geq 1 \end{cases}$$

Para el desarrollo de la demostración considerar que se tiene la proposición

$$p(n) = \sum_{i=0}^n \binom{n}{i} = \binom{n}{0} + \binom{n}{1} + \binom{n}{2} + \cdots + \binom{n}{n-1} + \binom{n}{n} = 2^n, \text{ para } n \in \mathbb{N}$$

la proposición $p(n)$ tomará el valor verdadero (V_o) o falso (F_o) dependiendo de si se cumple o no la igualdad $\sum_{i=0}^n \binom{n}{i} = 2^n$. Ahora se consideran los tres pasos de la técnica por inducción matemática:

Paso base $n = 0$:

$$p(0) = \underbrace{\sum_{i=0}^0 \binom{0}{i} = \binom{0}{0} = \frac{0!}{0! \cdot 0!} = \frac{1}{1 \cdot 1} = 1}_{\text{resultado a partir de la sumatoria de términos}} = \underbrace{2^0 = 1}_{\text{resultado a partir de la fórmula}}$$

como se obtiene el mismo resultado en la sumatoria de términos y en la fórmula que es la solución de la sumatoria entonces la proposición $p(0)$ es verdadera y la demostración continua en el paso inductivo.

Paso inductivo $n = k$:

$$p(k) = \sum_{i=0}^k \binom{k}{i} = \binom{k}{0} + \binom{k}{1} + \binom{k}{2} + \cdots + \binom{k}{k-1} + \binom{k}{k} = 2^k,$$

se asume que la proposición $p(k)$ es verdadera, esto quiere decir, que se supone que se cumple la siguiente igualdad:

$$\binom{k}{0} + \binom{k}{1} + \binom{k}{2} + \cdots + \binom{k}{k-1} + \binom{k}{k} = 2^k.$$

Paso post-inductivo $n = k + 1$:

$$p(k+1) = \sum_{i=0}^{k+1} \binom{k+1}{i} = \binom{k+1}{0} + \binom{k+1}{1} + \cdots + \binom{k+1}{k} + \binom{k+1}{k+1} = 2^{k+1}$$

Apoyados en la definición recursiva se tiene que:

$$\begin{aligned} \binom{k+1}{0} &= \binom{k}{0} = 1 \\ \binom{k+1}{1} &= \binom{k}{0} + \binom{k}{1} \\ \binom{k+1}{2} &= \binom{k}{1} + \binom{k}{2} \\ \binom{k+1}{3} &= \binom{k}{2} + \binom{k}{3} \\ &\vdots \\ \binom{k+1}{k} &= \binom{k}{k-1} + \binom{k}{k} \\ \binom{k+1}{k+1} &= \binom{k}{k} = 1 \end{aligned}$$

de esta forma se tiene que:

$$\begin{aligned}
& \underbrace{\binom{k}{0}}_{\binom{k+1}{0}} + \underbrace{\left[\binom{k}{0} + \binom{k}{1} \right]}_{\binom{k+1}{1}} + \underbrace{\left[\binom{k}{1} + \binom{k}{2} \right]}_{\binom{k+1}{2}} + \cdots + \underbrace{\left[\binom{k}{k-1} + \binom{k}{k} \right]}_{\binom{k+1}{k}} + \underbrace{\binom{k}{k}}_{\binom{k+1}{k+1}} = 2^{k+1} \\
& \left[\binom{k}{0} + \binom{k}{0} \right] + \left[\binom{k}{1} + \binom{k}{1} \right] + \left[\binom{k}{2} + \binom{k}{2} \right] + \cdots + \left[\binom{k}{k} + \binom{k}{k} \right] = 2^{k+1} \\
& 2 \binom{k}{0} + 2 \binom{k}{1} + 2 \binom{k}{2} + \cdots + 2 \binom{k}{k} = 2^{k+1} \\
& 2 \left[\binom{k}{0} + \binom{k}{1} + \binom{k}{2} + \cdots + \binom{k}{k} \right] = 2^{k+1}
\end{aligned}$$

*Se reemplaza por su equivalente
en el paso inductivo*

$$2[2^k] = 2^{k+1}$$

$$2^{k+1} = 2^{k+1}$$

Se cumple la igualdad, por lo tanto la proposición $p(k+1)$ es verdadera, como se cumplen los tres pasos de técnica de demostración por inducción matemática entonces queda demostrada la validez de la solución de la sumatoria originalmente planteada.

Ejemplo 20:

Sea la siguiente función trabajada en el ejemplo 27 del capítulo 2 del Uso de Sucesiones y Sumatorias en Computación

función Campeón(n : Entero Positivo)

$r = 0$

Para $i = 0$ Hasta $n - 1$

Para $j = i + 1$ Hasta n

$r = r + j + 2^i$

Retornar r

Utilizando la equivalencia del pseudocódigo de programación, entre el ciclo de repetición *Para* y las sumatorias, se logró determinar que:

$$Campeón(n) = \sum_{i=0}^{n-1} \sum_{j=i+1}^n (j + 2^i)$$

al solucionar las sumatorias se obtiene que la función en pseudocódigo de programación, es equivalente a la función matemática:

$$Campeón(n) = \frac{n \cdot (n+1) \cdot (2n+1)}{6} + 2 \cdot (2^n - 1) - n, \text{ para } n \in \mathbb{Z}^+$$

Ahora se demostrará utilizando la técnica de inducción matemática que:

$$\sum_{i=0}^{n-1} \sum_{j=i+1}^n (j + 2^i) = \frac{n \cdot (n+1) \cdot (2n+1)}{6} + 2 \cdot (2^n - 1) - n, \text{ para } n \in \mathbb{Z}^+$$

Para el desarrollo de la demostración considerar que se tiene la proposición:

$$p(n) = \sum_{i=0}^{n-1} \sum_{j=i+1}^n (j + 2^i) = \frac{n \cdot (n+1) \cdot (2n+1)}{6} + 2 \cdot (2^n - 1) - n$$

la proposición $p(n)$ tomará el valor verdadero (V_o) o falso (F_o) dependiendo de si se cumple o no la igualdad $\sum_{i=0}^{n-1} \sum_{j=i+1}^n (j + 2^i) = \frac{n \cdot (n+1) \cdot (2n+1)}{6} + 2 \cdot (2^n - 1) - n$.

Ahora se consideran los tres pasos de la técnica por inducción matemática:

Paso base $n = 1$:

$$\begin{aligned} p(1) &= \sum_{i=0}^{1-1} \sum_{j=i+1}^1 (j + 2^i) = \sum_{i=0}^0 \sum_{j=i+1}^1 (j + 2^i) = \sum_{j=1}^1 (j + 2^0) = 1 + 2^0 = 2 \\ &\quad \underbrace{\hspace{15em}}_{\text{resultado a partir de las sumatorias de términos}} \\ &= \underbrace{\frac{1 \cdot (1+1) \cdot (2(1) + 1)}{6} + 2 \cdot (2^1 - 1) - 1}_{\text{resultado a partir de la fórmula}} = 1 + 2 - 1 = 2 \end{aligned}$$

como se obtiene el mismo resultado en las sumatorias de términos y en la fórmula que es la solución de las sumatorias entonces la proposición $p(1)$ es verdadera y la demostración continua en el paso inductivo.

Paso inductivo $n = k$:

$$p(k) = \sum_{i=0}^{k-1} \sum_{j=i+1}^k (j + 2^i) = \frac{k \cdot (k+1) \cdot (2k+1)}{6} + 2 \cdot (2^k - 1) - k, \text{ se asume que la proposición } p(k) \text{ es verdadera.}$$

Paso post-inductivo $n = k + 1$:

$$\begin{aligned} p(k+1) &= \sum_{i=0}^k \sum_{j=i+1}^{k+1} (j + 2^i) \\ &= \sum_{i=0}^{k-1} \sum_{j=i+1}^{k+1} (j + 2^i) + \sum_{i=k}^k \sum_{j=i+1}^{k+1} (j + 2^i) \\ &= \sum_{i=0}^{k-1} \sum_{j=i+1}^{k+1} (j + 2^i) + \sum_{j=k+1}^{k+1} (j + 2^k) \\ &= \sum_{i=0}^{k-1} \sum_{j=i+1}^{k+1} (j + 2^i) + (k + 1 + 2^k) \end{aligned}$$

$$\begin{aligned}
&= \sum_{i=0}^{k-1} \left(\sum_{j=i+1}^k (j + 2^i) + \sum_{j=k+1}^{k+1} (j + 2^i) \right) + (k + 1 + 2^k) \\
&= \sum_{i=0}^{k-1} \sum_{j=i+1}^k (j + 2^i) + \sum_{i=0}^{k-1} \sum_{j=k+1}^{k+1} (j + 2^i) + (k + 1 + 2^k) \\
&= \sum_{i=0}^{k-1} \sum_{j=i+1}^k (j + 2^i) + \sum_{i=0}^{k-1} (k + 1 + 2^i) + (k + 1 + 2^k) \\
&= \sum_{i=0}^{k-1} \sum_{j=i+1}^k (j + 2^i) + \sum_{i=0}^{k-1} (k + 1) + \sum_{i=0}^{k-1} (2^i) + (k + 1 + 2^k) \\
&= \sum_{i=0}^{k-1} \sum_{j=i+1}^k (j + 2^i) + k \cdot (k + 1) + (2^k - 1) + (k + 1) + 2^k \\
&= \sum_{i=0}^{k-1} \sum_{j=i+1}^k (j + 2^i) + k \cdot (k + 1) + (k + 1) + 2^k + (2^k - 1) \\
&= \sum_{i=0}^{k-1} \sum_{j=i+1}^k (j + 2^i) + (k + 1) \cdot (k + 1) + 2^k + 2^k - 1 \\
&= \sum_{i=0}^{k-1} \sum_{j=i+1}^k (j + 2^i) + (k + 1)^2 + 2^k \cdot (1 + 1) - 1 \\
&= \sum_{i=0}^{k-1} \sum_{j=i+1}^k (j + 2^i) + (k + 1)^2 + 2^k \cdot (2) - 1 \\
&= \underbrace{\sum_{i=0}^{k-1} \sum_{j=i+1}^k (j + 2^i)}_{\text{Se reemplaza por su equivalente}} + (k + 1)^2 + 2^{k+1} - 1
\end{aligned}$$

Se reemplaza por su equivalente

en el paso inductivo

$$\begin{aligned}
&= \frac{k \cdot (k + 1) \cdot (2k + 1)}{6} + 2 \cdot (2^k - 1) - k + (k + 1)^2 + 2^{k+1} - 1 \\
&= \frac{k \cdot (k + 1) \cdot (2k + 1)}{6} + 2^{k+1} - 2 + (k + 1)^2 + 2^{k+1} - k - 1 \\
&= \frac{k \cdot (k + 1) \cdot (2k + 1)}{6} + (k + 1)^2 + 2^{k+1} + 2^{k+1} - 2 - (k + 1) \\
&= \left(\frac{k + 1}{6} \right) \cdot \left(k \cdot (2k + 1) + 6 \cdot (k + 1) \right) + 2 \cdot 2^{k+1} - 2 - (k + 1) \\
&= \left(\frac{k + 1}{6} \right) \cdot (2k^2 + k + 6k + 6) + 2 \cdot (2^{k+1} - 1) - (k + 1) \\
&= \left(\frac{k + 1}{6} \right) \cdot (2k^2 + 7k + 6) + 2 \cdot (2^{k+1} - 1) - (k + 1)
\end{aligned}$$

$$\begin{aligned}
&= \left(\frac{k+1}{6} \right) \cdot (2k^2 + 3k + 4k + 6) + 2 \cdot (2^{k+1} - 1) - (k + 1) \\
&= \left(\frac{k+1}{6} \right) \cdot \left((k+2) \cdot (2k+3) \right) + 2 \cdot (2^{k+1} - 1) - (k + 1) \\
&= \frac{(k+1) \cdot (k+2) \cdot (2k+3)}{6} + 2 \cdot (2^{k+1} - 1) - (k + 1) \\
&= \frac{(k+1) \cdot (k+1+1) \cdot (2k+2+1)}{6} + 2 \cdot (2^{k+1} - 1) - (k + 1) \\
&= \frac{(k+1) \cdot ((k+1)+1) \cdot (2 \cdot (k+1)+1)}{6} + 2 \cdot (2^{k+1} - 1) - (k + 1)
\end{aligned}$$

El resultado anterior se obtuvo al identificar y reemplazar en el caso post-inductivo el caso inductivo, donde lo que se obtiene como resultado, es la fórmula que se está demostrando que es la solución de las sumatorias evaluada en $n = k + 1$. De esta manera se cumple la igualdad y la proposición $p(k + 1)$ es verdadera. Al cumplirse los tres pasos de la técnica de demostración por inducción matemática entonces queda demostrada la validez de la solución que se obtuvo para la función *Campeón*.

3.6. Ejercicios

1. Probar o refutar cada uno de los siguientes ítems utilizando alguno de los métodos de demostración:
 - a.) La suma de dos números enteros pares es un entero par.
 - b.) La suma de dos números enteros impares es un entero par.
 - c.) La suma de un número entero impar con un número entero par es un número entero impar.
 - d.) Si el producto de dos números enteros es par, entonces alguno de los dos números que se esta multiplicando es par.
 - e.) Si el producto de dos números enteros es impar, entonces los dos números que se esta multiplicando son impares.
 - f.) El cuadrado de todo número entero es un número entero no negativo.
 - g.) Si el cuadrado de n no es divisible por 2 entonces n no es divisible por 2.
 - h.) Si un número entero es divisible por 4 entonces es divisible por 2.
 - i.) Si n es un entero positivo, entonces n es par si y únicamente si $7n + 4$ es par.
 - j.) La suma de cualquier número entero n con n^2 es par.
 - k.) Si n es un número entero y $n^3 + 1$ es impar, entonces n es par.
 - l.) Si n es un número entero y $n^3 + 5$ es par, entonces n es impar.

m.) Si n es un número entero y $3n + 2$ es par, entonces n es par.

$$\text{n.) } \binom{n}{r} + 2\binom{n}{r-1} + \binom{n}{r-2} = \binom{n+2}{r}, \text{ para } n \geq r \geq 2$$

$$\text{ñ.) } \binom{n}{0} + \binom{n+1}{1} + \binom{n+2}{2} + \cdots + \binom{n+r-1}{r-1} + \binom{n+r}{r} = \binom{n+r+1}{r}$$

donde n, r son números enteros positivos.

$$\begin{array}{l} \text{o.) } p \\ p \vee q \\ q \rightarrow (r \rightarrow s) \\ t \rightarrow r \\ \hline \therefore \neg s \rightarrow \neg t \end{array}$$

$$\begin{array}{l} \text{p.) } p \rightarrow q \\ q \rightarrow s \\ r \rightarrow \neg s \\ \neg p \otimes r \\ \hline \therefore \neg p \end{array}$$

q.) La suma de 4 enteros positivos consecutivos cualquiera es divisible por 4.

r.) La suma de 5 enteros positivos consecutivos cualquiera es divisible por 9.

2. Probar o refutar cada uno de los siguientes ítems utilizando el método de demostración por casos:

- a.) El producto de cualesquiera 3 enteros positivos consecutivos es divisible por 6.
- b.) El producto de cualesquiera 4 enteros positivos consecutivos es divisible por 12.
- c.) La diferencia entre los cuadrados de dos números enteros impares es divisible por 8. La demostración se tiene que cumplir para cualquier pareja de números enteros impares.
- d.) $n(n^2 + 5)$ es divisible por 3, para $n \in \mathbb{Z}^+, n \geq 1$
- e.) El cuadrado de cualquier número entero positivo finaliza con un 0, 1, 4, 5, 6 o 9. (Ayuda: Sea $n = 10k + j$ donde $j = 0, 1, \dots, 9$). Probar si es necesario cada uno de los diez casos y concluir.
- f.) El cubo de cualquier número entero positivo finaliza con un 0, 1, 2, 4, 5, 6, 7 u 8. (Ayuda: Sea $n = 10k + j$ donde $j = 0, 1, \dots, 9$). Probar si es necesario cada uno de los diez casos y concluir.
- g.) El cuadrado de todo número entero, es un múltiplo de 3, ó difiere de un múltiplo de 3 en 1.

h.) El cuadrado de todo número entero, es un múltiplo de 4, ó difiere de un múltiplo de 4 en 1.

3. Probar o refutar cada uno de los siguientes ítems utilizando el método de demostración por Inducción Matemática:

a.) $1(2) + 2(3) + 3(4) + 4(5) + \dots + n(n+1) = \frac{n(n+1)(n+2)}{3}$, para $n \in \mathbb{Z}^+$, $n \geq 1$.

b.) $0 + 3 + 8 + \dots + (n^2 - 1) = \frac{n(2n+5)(n-1)}{6}$, para $n \in \mathbb{Z}^+$, $n \geq 1$.

c.) $0 + 7 + 26 + \dots + (n^3 - 1) = \frac{n(n(n+1)^2 - 4)}{4}$, para $n \in \mathbb{Z}^+$, $n \geq 1$.

d.) $1 \cdot \frac{2}{3} + 2 \cdot \frac{5}{3} + \dots + n \cdot (n - \frac{1}{3}) = \frac{n^2(n+1)}{3}$, para $n \in \mathbb{Z}^+$, $n \geq 1$.

e.) $1 \cdot \frac{1}{4} + 4 \cdot \frac{5}{4} + 9 \cdot \frac{9}{4} + \dots + n^2 \cdot (n - \frac{3}{4}) = \frac{n(n+1)(2n^2-1)}{8}$, para $n \in \mathbb{Z}^+$, $n \geq 1$.

f.) $3 + 6 + 20 + \dots + (n(n!) + 2) = (n+1)! + 2n - 1$, para $n \in \mathbb{Z}^+$, $n \geq 1$.

g.) $1^2 + 3^2 + 5^2 + \dots + (2n-1)^2 = \frac{n(2n-1)(2n+1)}{3}$, para $n \in \mathbb{Z}^+$, $n \geq 1$.

h.) $3^2 + 4^2 + 5^2 + 6^2 + \dots + (n+2)^2 = \frac{(n+2)(n+3)(2n+5)}{6} - 5$, para $n \in \mathbb{Z}^+$, $n \geq 1$.

i.) $1^2 + 4^2 + 7^2 + 10^2 + \dots + (3n-2)^2 = \frac{n[3n(2n-1) - 1]}{2}$, para $n \in \mathbb{Z}^+$, $n \geq 1$.

j.) $1 \cdot 2^1 + 2 \cdot 2^2 + 3 \cdot 2^3 + 4 \cdot 2^4 + \dots + n \cdot 2^n = (n-1) \cdot 2^{n+1} + 2$, para $n \in \mathbb{Z}^+$, $n \geq 1$.

k.) $\frac{3}{1(3)} + \frac{3}{3(5)} + \frac{3}{5(7)} + \frac{3}{7(9)} + \dots + \frac{3}{(2n-1)(2n+1)} = \frac{3n}{2n+1}$, para $n \in \mathbb{Z}^+$, $n \geq 1$.

l.) $\frac{1}{1(2)} + \frac{1}{2(3)} + \frac{1}{3(4)} + \frac{1}{4(5)} + \dots + \frac{1}{n(n+1)} = \frac{n}{n+1}$, para $n \in \mathbb{Z}^+$, $n \geq 1$.

m.) $11^n - 6$ es divisible por 5, para $n \in \mathbb{Z}^+$, $n \geq 1$.

n.) $3^{2n} + 7$ es divisible por 8, para $n \in \mathbb{Z}^+$, $n \geq 1$.

ñ.) $n^3 - n$ es divisible por 3, para $n \in \mathbb{Z}^+$, $n \geq 1$.

o.) $Misterio(n) = \sum_{i=1}^{n-1} \sum_{j=i+1}^n \sum_{k=1}^j 1 = \frac{(n-1) \cdot n \cdot (n+1)}{3}$, para $n \in \mathbb{Z}^+$.

p.) $Aloha(n) = \sum_{i=1}^{n-1} \sum_{j=i+1}^n 1 = \frac{(n-1) \cdot n}{2}$, para $n \in \mathbb{Z}^+$.

q.) $Harehare(n) = \sum_{j=1}^n \sum_{k=j}^n 2^k = (n-1) \cdot 2^{n+1} + 2$, para $n \in \mathbb{Z}^+$.

3.7. Preguntas tipo ECAES

1. Para demostrar que una propiedad $P(n)$ es verdadera para cualquier número natural (entero no negativo), basta probar que se cumple la siguiente proposición⁴

- A.) $P(0)$ y, para $k \geq 0$: $P(k) \Rightarrow P(k+1)$
- B.) $P(0), P(1)$ y, para $k > 0$: $P(k) \Rightarrow P(k+2)$
- C.) $P(0)$ y, para $k \geq 0$: $P(k+1) \Rightarrow P(k)$
- D.) $P(0), P(1), P(2), \dots, P(k)$ implican $P(k)$, para $k \geq 0$
- E.) Para $k \geq 0$: $P(k+1) \Rightarrow P(k)$

2. Cuál es el resultado de la sumatoria⁵: $5^3 + 5^4 + 5^5 + \dots + 5^n$?

- A.) $\frac{5^3(5^{n-2} - 1)}{4}$
- B.) $\frac{5^3(5^{n+1} - 1)}{4}$
- C.) $\frac{5^{n+1} - 1}{4}$
- D.) $\frac{5^3(5^n - 1)}{4}$
- E.) $5^3(5^{n-2} - 1)$

3. Cuál es el resultado de la sumatoria: $2^0 - 2^1 + 2^2 - 2^3 + 2^4 - 2^5 + \dots + (-1)^n * 2^n$?

- A.) $\frac{-1 + (-2)^{n+1}}{-3}$
- B.) $\frac{(-2)^{n+1} - 1}{3}$
- C.) $\frac{(-2)^{n+1} + 1}{3}$
- D.) $(-2)^{n+1} - 1$
- E.) $\frac{1 + (-1)^n(2)^{n+1}}{3}$

4. Cuál es el resultado de la sumatoria: $3^0 - 3^1 + 3^2 - 3^3 + 3^4 - 3^5 + \dots + (-1)^n * 3^n$?

- A.) $\frac{(-3)^{n+1} - 1}{2}$
- B.) $\frac{(-3)^{n+1} + 1}{4}$

⁴Enunciado textual de la pregunta 15 del cuadernillo de la segunda sesión del ECAES de Ingeniería de Sistemas del año 2004 [MIA2004].

⁵Las preguntas desde la 2 hasta la 7 son del autor.

- C.) $\frac{(-3)^{n+1} + 1}{2}$
D.) $\frac{(-1)^n(3)^{n+1} + 1}{4}$
E.) $\frac{(-1)^{n+1}(3)^{n+1} + 1}{4}$

5. Cuál es el resultado de la suma de términos: $1(2) + 2(3) + 3(4) + 4(5) + \dots + n(n+1)$?

- A.) $\frac{n(n+1)(2n+1)}{3}$
B.) $\frac{n(n+1)(2n+1)}{6}$
C.) $\frac{n(n+1)(2n-1)}{3}$
D.) $\frac{n(n+1)(n+2)}{3}$
E.) $\frac{n(n+1)(n+2)}{6}$

6. Cuál es el resultado de la suma de términos: $1(3) + 2(4) + 3(5) + 4(6) + \dots + n(n+2)$?

- A.) $\frac{n(n+2)(2n+1)}{3}$
B.) $\frac{n(n+1)(2n+7)}{6}$
C.) $\frac{n(n+1)(2n-1)}{3}$
D.) $\frac{n(n+1)(n+2)}{3}$
E.) $\frac{n(n+1)(n+2)}{6}$

7. Cuál es el resultado de la sumatoria: $\frac{1}{1(2)} + \frac{1}{2(3)} + \frac{1}{3(4)} + \frac{1}{4(5)} + \dots + \frac{1}{n(n+1)}$?

- A.) $\frac{1}{n+1}$
B.) $\frac{4}{n+1}$
C.) $\frac{n}{n+1}$
D.) $\frac{2n}{n+1}$
E.) n

Capítulo 4

Relaciones de recurrencia

A menudo es posible encontrar relaciones entre los elementos de una sucesión. Estas relaciones se llaman relaciones de recurrencia.

Una relación de recurrencia para una sucesión $a_0, a_1, a_2, \dots, a_n$ es una ecuación que relaciona a_n con alguno (o algunos) de sus antecesores $a_0, a_1, a_2, \dots, a_{n-1}$ y la suma o multiplicación de alguna cantidad.

Ejemplo 1:

En el ejemplo 9 del Capítulo 2 de Sucesiones y Sumatorias se pide que se genere una fórmula para calcular el n -ésimo término de la sucesión que tiene los primeros 10 términos siguientes: 3, 9, 15, 21, 27, 33, 39, 45, 51 y 57. La fórmula que se obtuvo en este ejemplo es:

$$\begin{aligned} S_1 &= 3 \\ S_n &= S_{n-1} + 6, \text{ para } n \geq 2, n \in \mathbb{Z}^+ \end{aligned}$$

esta fórmula es una relación de recurrencia, con la cual se indica que el término ubicado en la posición n de la sucesión se obtiene al relacionar el término que se encuentra en la sucesión en la posición $n - 1$ con la suma del número 6.

¿Cuántos llamados recursivos son necesarios en la fórmula anterior para calcular el elemento que se encuentra en la posición un millón de la sucesión?

Son necesarios un millón de llamados recursivos, y en cada llamado recursivo a excepción del caso base, es realizada una suma. La fórmula recursiva o relación de recurrencia anterior, es correcta para calcular el n -ésimo término de la sucesión, pero, tiene un costo computacional muy alto; por este motivo es fundamental determinar fórmulas sin recursividad para generar el n -ésimo término de la sucesión. La solución de relaciones de recurrencia permite generar a partir de la relación de recurrencia una fórmula sin recursividad, que permite generar el mismo valor n -ésimo de la sucesión.

La fórmula $S_n = 3 + 6 * (n - 1)$, para $n \in \mathbb{Z}^+$, es la solución de la relación de recurrencia, y para calcular el elemento ubicado en la posición n de la sucesión, sólo necesita realizar tres operaciones (una resta, una multiplicación y una suma) independientemente del valor de n .

En los ejercicios de la siguiente sección se presentan las relaciones de recurrencia que permiten generar de forma recursiva el n -ésimo término de una sucesión y se pide que utilizando el método de iteración se obtenga una fórmula que permita generar el mismo n -ésimo término de la sucesión sin necesidad de consultar términos previos en la sucesión, para de ésta forma evitar el costo computacional de la recursividad.

4.1. Método de Iteración

En la serie de ejercicios de esta sección se utilizará el Método de Iteración para resolver las relaciones de recurrencia de primer orden no homogéneas, el método consiste en comenzar en el caso base de la relación de recurrencia y utilizarlo para definir sin recursividad el caso que sigue después del caso base, y así sucesivamente se itera tantas veces como sea necesario, hasta lograr determinar cuál es la sumatoria o sumatorias “ocultas” que sirven para solucionar la relación de recurrencia sin utilizar recursividad.

Después de tener la sumatoria o sumatorias para la relación de recurrencia evaluada en un valor n , estas (o esta) son resueltas y de esta forma se obtiene la solución de la relación de recurrencia.

Por último el método de demostración por inducción matemática puede ser utilizado para ratificar o refutar que la solución de la relación de recurrencia es correcta.

Ejemplo 2:

Resolver la siguiente relación de recurrencia:

$$P(1) = 1$$

$$P(n) = P(n - 1) + n, \text{ para } n > 1, n \in \mathbb{Z}^+$$

Utilizando el método de iteración se tiene:

$$P(1) = 1$$

$$P(2) = P(1) + 2 = 1 + 2$$

no dar el resultado de $1 + 2$ sino dejar indicada la suma de términos para no desaparecer la sumatoria.

$$P(3) = P(2) + 3 = 1 + 2 + 3$$

$$P(4) = P(3) + 4 = 1 + 2 + 3 + 4$$

$$P(5) = P(4) + 5 = 1 + 2 + 3 + 4 + 5$$

Se itera tantas como se considere necesario hasta que se identifique cual es la sumatoria oculta. En estos momentos debe ser evidente que:

$$P(n) = 1 + 2 + 3 + 4 + \dots + n$$

de esta forma se detecta que la sumatoria oculta que resuelve la relación de recurrencia sin recursividad es $\sum_{i=1}^n i = 1 + 2 + \dots + n$

la cual tiene solución $\frac{n(n+1)}{2}$. Por lo tanto $P(n) = \sum_{i=1}^n i = \frac{n(n+1)}{2}$. De esta forma la solución de la relación de recurrencia es $P(n) = \frac{n(n+1)}{2}$ para $n \geq 1$

Ahora se va a utilizar el método de demostración por inducción matemática para ratificar que la solución obtenida de la relación de recurrencia es correcta.

Caso base $n = 1$

n toma el valor de 1 porque este es el valor con el cuál termina la recursividad en la relación de recurrencia, o dicho en otras palabras, para $n=1$, esta definido el caso base de la relación de recurrencia.

$$\underbrace{P(1) = 1}_{\text{Caso base de la R.R.}} = \underbrace{\frac{1(2)}{2}}_{\text{Solución de la relación de recurrencia evaluada en 1}} = 1$$

Como se cumple la igualdad entre el caso base de la relación de recurrencia y la solución de la relación de recurrencia evaluada en $n = 1$, entonces la demostración continua en el caso inductivo.

Caso inductivo $n=k$:

Se asume como cierta la solución de la relación de recurrencia evaluada en k , donde $k \in \mathbb{Z}^+$ para $k > 1$

$$P(k) = \frac{k(k+1)}{2}$$

Caso post-inductivo $n = k + 1$:

en este caso se recuerda el paso recursivo de la relación de recurrencia donde

$P(n) = P(n-1) + n$, reemplazando n por $k+1$ se tiene:

$$P(k+1) = P(k+1-1) + k+1$$

$$P(k+1) = P(k) + (k+1)$$

donde $P(k+1)$ y $P(k)$ se reemplazan por su equivalente en el caso inductivo evaluado en $k+1$ y k respectivamente

$$\frac{(k+1)(k+1+1)}{2} = \frac{k(k+1)}{2} + (k+1)$$

$$\begin{aligned} \frac{(k+1)(k+2)}{2} &= (k+1) \left[\frac{k}{2} + 1 \right] \\ &= (k+1) \left[\frac{k}{2} + \frac{2}{2} \right] \\ &= (k+1) \left[\frac{k+2}{2} \right] \\ &= \frac{(k+1)(k+2)}{2} \end{aligned}$$

Como efectivamente se llegó a la igualdad, entonces, se ratifica que la solución de la relación de recurrencia obtenida por el método de iteración es correcta.

Ejemplo 3:

Resolver la siguiente relación de recurrencia:

$$P(1) = 2$$

$$P(n) = P(n-1) + n \cdot 2^n, \text{ para } n > 1$$

Utilizando el método de iteración se tiene:

$$P(1) = 2$$

$$P(2) = P(1) + 2 \cdot 2^2 = 2 + 2 \cdot 2^2 = 1 \cdot 2^1 + 2 \cdot 2^2$$

recordar que no se calculan las potencias ni se hacen las multiplicaciones, ni se hacen las sumas porque se desaparecería la sumatoria “oculta” que está debajo de la relación de recurrencia. Retomando el método iterativo se tiene:

$$P(3) = P(2) + 3 \cdot 2^3 = 1 \cdot 2^1 + 2 \cdot 2^2 + 3 \cdot 2^3$$

$$P(4) = P(3) + 4 \cdot 2^4 = 1 \cdot 2^1 + 2 \cdot 2^2 + 3 \cdot 2^3 + 4 \cdot 2^4$$

⋮

$$P(n) = 1 \cdot 2^1 + 2 \cdot 2^2 + 3 \cdot 2^3 + \dots + n \cdot 2^n$$

lo que se hizo en el paso anterior fue dejar de iterar porque ya se identificó la sumatoria y generalizar esta cuando la relación de recurrencia P es evaluada en “ n ”.

Como $P(n) = \sum_{i=1}^n i \cdot 2^i$ entonces la solución de la relación de recurrencia es la solución de la sumatoria. Tomando la solución de dicha sumatoria de la sección de “Sumatorias Especiales” se tiene que $P(n) = \sum_{i=1}^n i \cdot 2^i = (n-1) \cdot 2^{n+1} + 2$, por lo tanto la solución de la relación de recurrencia es $P(n) = (n-1) \cdot 2^{n+1} + 2$, para $n \geq 1$.

Ahora se utilizará el método de demostración por inducción matemática para ratificar la validez de la solución de la relación de recurrencia, por lo tanto se tiene:

Caso base $n = 1$:

$$\underbrace{P(1) = 2}_{\text{caso base de la R.R.}} = \underbrace{(1-1) \cdot 2^{1+1} + 2 = 0 \cdot 2^2 + 2 = 0 + 2 = 2}_{\text{Solución de la relación de recurrencia evaluada en 1}}$$

Como se cumple la igualdad, entonces sigue la demostración con el caso inductivo.

Caso inductivo $n=k$:

Se asume como cierta la solución de la relación de recurrencia evaluada en k , donde $k \in \mathbb{Z}^+$ para $k > 1$

$$P(k) = (k-1) \cdot 2^{k+1} + 2$$

Caso post-inductivo $n = k+1$:

Recordar el paso recursivo de la relación de recurrencia:

$$P(n) = P(n-1) + n \cdot 2^n, \text{ reemplazando } n \text{ por } k+1 \text{ se tiene:}$$

$$P(k+1) = P(k+1-1) + (k+1) \cdot 2^{k+1}$$

$$P(k+1) = P(k) + (k+1) \cdot 2^{k+1}$$

donde $P(k+1)$ y $P(k)$ se reemplazan por su equivalente en el caso inductivo evaluado en $k+1$ y k respectivamente.

$$\underbrace{(k+1-1) \cdot 2^{k+1+1} + 2}_{P(k+1)} = \underbrace{(k-1) \cdot 2^{k+1} + 2}_{P(k)} + (k+1) \cdot 2^{k+1}$$

$$\begin{aligned} k \cdot 2^{k+2} + 2 &= (k-1 + k+1) \cdot 2^{k+1} + 2 \\ &= (2k) \cdot 2^{k+1} + 2 \\ &= k \cdot 2^{k+2} + 2 \end{aligned}$$

Como se llegó a una igualdad, entonces, se ratifica que la solución de la relación de recurrencia obtenida por el método de iteración es correcta.

Ejemplo 4:

Resolver la siguiente relación de recurrencia:

$$P(1) = 1$$

$$P(n) = 2P(n-1) + 1, \text{ para } n > 1$$

Por el método de iteración se tiene:

$$P(1) = 1$$

$$P(2) = 2P(1) + 1 = 2(1) + 1 = 2^1 + 2^0$$

$$P(3) = 2P(2) + 1 = 2(2^1 + 2^0) + 1 = 2^2 + 2^1 + 2^0$$

$$P(4) = 2P(3) + 1 = 2(2^2 + 2^1 + 2^0) + 1 = 2^3 + 2^2 + 2^1 + 2^0$$

\vdots

$$P(n) = 2^{n-1} + 2^{n-2} + \dots + 2^1 + 2^0$$

$$P(n) = 2^0 + 2^1 + \dots + 2^{n-1}$$

Como $P(n) = \sum_{i=0}^{n-1} 2^i$ entonces la solución de la relación de recurrencia es la solución de la suma de términos de la serie geométrica con primer término $a = 2^0 = 1$, razón constante $r = 2$ y potencia más grande igual a $n - 1$ tomando la solución de dicha sumatoria de la sección de “Sumatorias Especiales” se tiene que

$$P(n) = \sum_{i=0}^{n-1} 2^i = \frac{a \cdot r^{(\text{potencia más grande})+1} - a}{r - 1} = \frac{1 \cdot 2^{(n-1)+1} - 1}{2 - 1} = 2^n - 1$$

por lo tanto la solución de la relación de recurrencia es $P(n) = 2^n - 1$, para $n \geq 1$.

Si se quiere ratificar o refutar la solución obtenida para la relación de recurrencia, entonces se puede utilizar el método de demostración por inducción matemática.

Ejemplo 5:

Resolver la siguiente relación de recurrencia:

$$P(1) = 2$$

$$P(n) = 2P(n-1) + 3^n - 1, \text{ para } n \geq 2$$

La diferencia de este ejemplo con los anteriores, radica en la cantidad de términos independientes que se encuentran en el caso recursivo, donde se tiene que al llamado

recursivo se le suma $3^n - 1$. Cada uno de los términos independientes genera su propia suma de términos.

Utilizando el método de iteración se tiene:

$$P(1) = 2$$

$$P(2) = 2P(1) + 3^2 - 1 = 2(2) + 3^2 - 1 = 2^2 + 3^2 - 1$$

$$P(3) = 2P(2) + 3^3 - 1 = 2(2^2 + 3^2 - 1) + 3^3 - 1$$

$$= (2^3 + 2 \cdot 3^2 - 2) + 3^3 - 1$$

$$= (2^3 + 2^1 \cdot 3^2 - 2^1) + 2^0 \cdot 3^3 - 2^0$$

$$= (2^3) + (2^1 \cdot 3^2 + 2^0 \cdot 3^3) - (2^1 + 2^0)$$

$$P(4) = 2P(3) + 3^4 - 1 = 2\left((2^3) + (2^1 \cdot 3^2 + 2^0 \cdot 3^3) - (2^1 + 2^0)\right) + 3^4 - 1$$

$$= \left((2^4) + (2^2 \cdot 3^2 + 2^1 \cdot 3^3) - (2^2 + 2^1)\right) + 2^0 \cdot 3^4 - 2^0$$

$$= (2^4) + (2^2 \cdot 3^2 + 2^1 \cdot 3^3 + 2^0 \cdot 3^4) - (2^2 + 2^1 + 2^0)$$

⋮

$$P(n) = (2^n) + (2^{n-2} \cdot 3^2 + 2^{n-3} \cdot 3^3 + \dots + 2^0 \cdot 3^n) - (2^{n-2} + 2^{n-3} + \dots + 2^0)$$

$$= (2^n) + 3^2(2^{n-2} \cdot 3^0 + 2^{n-3} \cdot 3^1 + \dots + 2^0 \cdot 3^{n-2}) - (2^0 + 2^1 + \dots + 2^{n-2})$$

$$P(n) = \underbrace{(2^n)}_{\text{término generado por el caso base de la relación de recurrencia}} + 3^2 \underbrace{(2^{n-2} \cdot 3^0 + 2^{n-3} \cdot 3^1 + \dots + 2^0 \cdot 3^{n-2})}_{\text{Sumatoria A}} - \underbrace{(2^0 + 2^1 + \dots + 2^{n-2})}_{\text{Sumatoria B}}$$

Sumatoria A:

La sumatoria A es una serie geométrica con primer término $a = 2^{n-2}$, razón constante $r = \frac{3}{2}$ y potencia más grande igual a $n - 2$, reemplazando en la formula se tiene:

$$\frac{a \cdot r^{(\text{potencia más grande})+1} - a}{r - 1} = \frac{2^{n-2} \cdot \left(\frac{3}{2}\right)^{(n-2)+1} - 2^{n-2}}{\frac{3}{2} - 1}$$

$$\begin{aligned}
&= \frac{2^{n-2} \cdot \left(\frac{3}{2}\right)^{n-1} - 2^{n-2}}{\frac{3}{2} - \frac{2}{2}} \\
&= \frac{2^{n-2} \cdot \frac{3^{n-1}}{2^{n-1}} - 2^{n-2}}{\frac{3-2}{2}} \\
&= \frac{2^{n-2} \cdot \frac{3^{n-1}}{2 \cdot 2^{n-2}} - 2^{n-2}}{\frac{1}{2}} \\
&= \frac{\frac{3^{n-1}}{2} - \frac{2}{2} \cdot 2^{n-2}}{\frac{1}{2}} \\
&= \frac{\frac{3^{n-1} - 2^{n-1}}{2}}{\frac{1}{2}} \\
&= 3^{n-1} - 2^{n-1}
\end{aligned}$$

Sumatoria B:

La sumatoria B es una serie geométrica con primer término $a = 2^0 = 1$, razón constante $r = 2$ y potencia más grande igual a $n - 2$, reemplazando en la formula se tiene:

$$\begin{aligned}
\frac{a \cdot r^{(potencia \text{ más grande})+1} - a}{r - 1} &= \frac{1 \cdot 2^{(n-2)+1} - 1}{2 - 1} \\
&= \frac{1 \cdot 2^{n-1} - 1}{1} \\
&= 2^{n-1} - 1
\end{aligned}$$

Solución Relación de recurrencia:

$$\begin{aligned}
P(n) &= (\text{término generado por el caso base de la relación de recurrencia}) \\
&\quad + 3^2 \cdot (\text{Solución Sumatoria A}) - (\text{Solución Sumatoria B})
\end{aligned}$$

$$\begin{aligned}
P(n) &= (2^n) + 3^2(3^{n-1} - 2^{n-1}) - (2^{n-1} - 1) \\
&= 2 \cdot 2^{n-1} + 3^2 \cdot 3^{n-1} - 3^2 \cdot 2^{n-1} - 2^{n-1} + 1 \\
&= 3^{n-1+2} + 2 \cdot 2^{n-1} - 9 \cdot 2^{n-1} - 2^{n-1} + 1
\end{aligned}$$

$$\begin{aligned}
&= 3^{n+1} + (2 - 9 - 1) \cdot 2^{n-1} + 1 \\
&= 3^{n+1} - 8 \cdot 2^{n-1} + 1 \\
&= 3^{n+1} - 2^3 \cdot 2^{n-1} + 1 \\
&= 3^{n+1} - 2^{n-1+3} + 1 \\
&= 3^{n+1} - 2^{n+2} + 1
\end{aligned}$$

Ejemplo 6:

Resolver la siguiente relación de recurrencia:

$$\begin{aligned}
P(1) &= 1 \\
P(n) &= 2P\left(\frac{n}{2}\right) + n, \text{ para } n = 2^m, m \in \mathbb{Z}^+.
\end{aligned}$$

El objetivo principal de este ejemplo es poner en evidencia que el llamado recursivo de la relación de recurrencia no siempre tiene que ser en términos de $n - 1$. Perfectamente el valor de n puede decrecer al dividir éste de forma constante por un mismo valor, en este ejemplo dicho valor es 2.

Para poder utilizar el método de iteración sobre esta relación de recurrencia es necesario primero hacer un cambio de variables.

Cambio de variables:

Como $n = 2^m$, entonces la relación de recurrencia original puede ser reescrita como:

$$\begin{aligned}
P(2^0) &= 2^0 \\
P(2^m) &= 2P\left(\frac{2^m}{2}\right) + 2^m, m \geq 1 \\
P(2^m) &= 2P(2^m \cdot 2^{-1}) + 2^m, m \geq 1 \\
P(2^m) &= 2P(2^{m-1}) + 2^m, m \geq 1
\end{aligned}$$

Método iterativo

$$\begin{aligned}
P(2^0) &= 2^0 \\
P(2^1) &= 2P(2^0) + 2^1 = 2^1 \cdot 2^0 + 2^1 = 2^1 + 2^1 \\
P(2^2) &= 2P(2^1) + 2^2 = 2[2^1 + 2^1] + 2^2 = 2^2 + 2^2 + 2^2 \\
P(2^3) &= 2P(2^2) + 2^3 = 2[2^2 + 2^2 + 2^2] + 2^3 = 2^3 + 2^3 + 2^3 + 2^3 \\
&\vdots \\
P(2^m) &= \underbrace{2^m + 2^m + 2^m + \dots + 2^m}_{m+1 \text{ veces}} = (m+1)2^m
\end{aligned}$$

De esta forma se halla la solución a la relación de recurrencia.

Ahora, después de obtener la solución de la relación de recurrencia es necesario utilizar algún mecanismo para ratificar o refutar la validez de la solución, por este motivo es normal que se utilice la técnica de demostración por inducción matemática para éste fin.

Prueba por inducción matemática:

Caso base $m = 0$:

$$\begin{aligned} P(2^0) &= 1 = (0 + 1)2^0 \\ &= (1)1 \\ &= 1 \end{aligned}$$

Como se cumple la igualdad entre el caso base de la relación de recurrencia y la solución de la relación de recurrencia evaluada en $m = 0$, entonces la demostración continua en el caso inductivo.

Caso inductivo $m = k$:

Se asume como cierto que: $P(2^k) = (k + 1)2^k$

Caso post-inductivo $m = k + 1$:

Recordar el caso recursivo de la relación de recurrencia:

$$P(2^m) = 2P(2^{m-1}) + 2^m$$

Al reemplazar m por $k + 1$ se tiene:

$$P(2^{k+1}) = 2P(2^k) + 2^{k+1}$$

Se reemplaza a $P(2^{k+1})$ y $P(2^k)$ por su equivalente en el caso inductivo.

$$\underbrace{((k + 1) + 1)2^{k+1}}_{P(2^{k+1})} = 2 \underbrace{[(k + 1)2^k]}_{P(2^k)} + 2^{k+1}$$

$$\begin{aligned} (k + 2)2^{k+1} &= (k + 1)2^{k+1} + 2^{k+1} \\ &= ((k + 1) + 1)2^{k+1} \\ &= (k + 2)2^{k+1} \end{aligned}$$

Se cumple la igualdad por lo tanto es correcta la solución de la relación de recurrencia.

Con respecto al cambio de variables se resolvió la relación de recurrencia y se demostró por inducción matemática que dicha solución es correcta, ahora lo que se va a hacer es presentar la solución de la relación de recurrencia con respecto a la variable original, para esto se debe recordar que $n = 2^m$, entonces $\lg_2 n = \lg_2 2^m$, $\lg_2 n = m$, $m = \lg_2 n$, reemplazando n y m en la solución de la relación de recurrencia $P(2^m) = (m + 1)2^m$, se tiene $P(n) = ((\lg_2 n) + 1)n$, $P(n) = n(1 + \lg_2 n)$, $P(n) = n + n \lg_2 n$.

Ejemplo 7:

Resolver la siguiente relación de recurrencia:

$$P(1) = 1$$

$$P(n) = 3P\left(\frac{n}{5}\right) + n, \text{ para } n = 5^m, m \in \mathbb{Z}^+.$$

En este ejemplo es necesario hacer primero el cambio de variables y reescribir la relación de recurrencia.

Cambio de variables:

Como $n = 5^m$, entonces la relación de recurrencia original puede ser reescrita de la siguiente forma:

$$P(5^0) = 5^0$$

$$P(5^m) = 3P\left(\frac{5^m}{5}\right) + 5^m, m \geq 1$$

$$P(5^m) = 3P(5^m \cdot 5^{-1}) + 5^m, m \geq 1$$

$$P(5^m) = 3P(5^{m-1}) + 5^m, m \geq 1$$

Método iterativo

$$P(5^0) = 5^0$$

$$P(5^1) = 3P(5^0) + 5^1 = 3^1 \cdot 5^0 + 3^0 \cdot 5^1$$

$$P(5^2) = 3P(5^1) + 5^2 = 3[3^1 \cdot 5^0 + 3^0 \cdot 5^1] + 5^2$$

$$= 3^2 \cdot 5^0 + 3^1 \cdot 5^1 + 3^0 \cdot 5^2$$

$$P(5^3) = 3P(5^2) + 5^3 = 3[3^2 \cdot 5^0 + 3^1 \cdot 5^1 + 3^0 \cdot 5^2] + 5^3$$

$$= 3^3 \cdot 5^0 + 3^2 \cdot 5^1 + 3^1 \cdot 5^2 + 3^0 \cdot 5^3$$

$$P(5^4) = 3P(5^3) + 5^4 = 3[3^3 \cdot 5^0 + 3^2 \cdot 5^1 + 3^1 \cdot 5^2 + 3^0 \cdot 5^3] + 5^4$$

$$= 3^4 \cdot 5^0 + 3^3 \cdot 5^1 + 3^2 \cdot 5^2 + 3^1 \cdot 5^3 + 3^0 \cdot 5^4$$

$$\vdots$$

$$P(5^m) = 3^m \cdot 5^0 + 3^{m-1} \cdot 5^1 + 3^{m-2} \cdot 5^2 + \dots + 3^1 \cdot 5^{m-1} + 3^0 \cdot 5^m$$

$$= \sum_{i=0}^m 3^m \cdot \left(\frac{5}{3}\right)^i$$

La Sumatoria que describe a $P(5^m)$ es una serie geométrica con primer término $a = 3^m$, razón constante $r = \frac{5}{3}$ y potencia más grande igual a m , reemplazando en la fórmula de la serie geométrica se tiene:

$$\begin{aligned} P(5^m) &= \left[\frac{3^m \cdot \left(\frac{5}{3}\right)^{m+1} - 3^m}{\frac{5}{3} - 1} \right] \\ &= 3^m \cdot \left[\frac{\frac{5^{m+1}}{3^{m+1}} - 1}{\frac{2}{3}} \right] = 3^m \cdot \left[\frac{\frac{5^{m+1} - 3^{m+1}}{3^{m+1}}}{\frac{2}{3}} \right] \\ &= 3^{m+1} \cdot \left[\frac{5^{m+1} - 3^{m+1}}{2 \cdot 3^{m+1}} \right] \\ &= \frac{5^{m+1} - 3^{m+1}}{2} \end{aligned}$$

De esta forma se halla la solución a la relación de recurrencia.

Para ratificar que es correcta la solución de la relación de recurrencia, entonces a continuación se utilizará la técnica de demostración por inducción matemática para éste fin.

Prueba por inducción matemática:

Caso base $m = 0$:

$$\begin{aligned} P(5^0) &= 1 = \frac{5^{0+1} - 3^{0+1}}{2} \\ &= \frac{5^1 - 3^1}{2} \\ &= \frac{5 - 3}{2} = \frac{2}{2} = 1 \end{aligned}$$

Como se cumple la igualdad entre el caso base de la relación de recurrencia y la solución de la relación de recurrencia evaluada en $m = 0$, entonces la demostración continua en el caso inductivo.

Caso inductivo $m = k$:

$$\text{Se asume como cierto que: } P(5^k) = \frac{5^{k+1} - 3^{k+1}}{2}$$

Caso post-inductivo $m = k + 1$:

Recordar el caso recursivo de la relación de recurrencia:

$$P(5^m) = 3P(5^{m-1}) + 5^m$$

Al reemplazar m por $k + 1$ se tiene:

$$P(5^{k+1}) = 3P(5^k) + 5^{k+1}$$

Se reemplaza a $P(5^{k+1})$ y $P(5^k)$ por su equivalente en el caso inductivo.

$$\begin{aligned} \underbrace{\frac{5^{k+2} - 3^{k+2}}{2}}_{P(5^{k+1})} &= 3 \underbrace{\left[\frac{5^{k+1} - 3^{k+1}}{2} \right]}_{P(5^k)} + 5^{k+1} \\ &= \frac{3 \cdot 5^{k+1} - 3 \cdot 3^{k+1}}{2} + \frac{2}{2} \cdot 5^{k+1} \\ &= \frac{3 \cdot 5^{k+1} - 3^{k+2} + 2 \cdot 5^{k+1}}{2} \\ &= \frac{3 \cdot 5^{k+1} + 2 \cdot 5^{k+1} - 3^{k+2}}{2} \\ &= \frac{5^{k+1}(3 + 2) - 3^{k+2}}{2} \\ &= \frac{5^{k+1}(5) - 3^{k+2}}{2} \\ &= \frac{5^{k+2} - 3^{k+2}}{2} \end{aligned}$$

Se cumple la igualdad por lo tanto es correcta la solución de la relación de recurrencia.

Ejemplo 8:

Sea la siguiente relación de recurrencia:

$$P(1) = 4$$

$$P(n) = 4P\left(\frac{n}{4}\right) + n^2 + 2, \text{ para } n = 4^m, m \in \mathbb{Z}^+.$$

El aporte fundamental de este ejemplo es que el caso base de la relación de recurrencia no siempre tiene que comenzar en uno, en este ejemplo comienza en cuatro (pudiéndose utilizar cualquier otro valor), y no se afecta en nada la utilización del método de iteración.

Ahora, primero se debe hacer el cambio de variables y reescribir la relación de recurrencia.

Cambio de variables:

Como $n = 4^m$, entonces la relación de recurrencia original puede ser reescrita de la siguiente forma:

$$P(4^0) = 4$$

$$P(4^m) = 4P\left(\frac{4^m}{4}\right) + (4^m)^2 + 2, m \geq 1$$

$$P(4^m) = 4P(4^m \cdot 4^{-1}) + (4^2)^m + 2, m \geq 1$$

$$P(4^m) = 4P(4^{m-1}) + 16^m + 2, m \geq 1$$

La cantidad de términos independientes en el caso recursivo de la relación de recurrencia indica cuantas sumatorias diferentes se presentan para solucionar la relación de recurrencia, en este ejemplo se generarán dos sumatorias, una en términos de 16^m y la otra en términos de 2, adicionalmente es normal que el término que se genera gracias al caso base de la relación de recurrencia no encaje en ninguna de las sumatorias, por este motivo se deja por aparte como si fuera una nueva sumatoria, pero de un solo término.

Método iterativo:

$$P(4^0) = 4$$

$$P(4^1) = 4P(4^0) + 16^1 + 2$$

$$= 4(4) + 16^1 + 2$$

$$= 4^2 + 16^1 + 2$$

$$P(4^2) = 4P(4^1) + 16^2 + 2$$

$$= 4(4^2 + 16^1 + 2) + 16^2 + 2$$

$$= 4^3 + 4^1 \cdot 16^1 + 4 \cdot 2 + 16^2 + 2$$

$$= 4^3 + 4^1 \cdot 16^1 + 4 \cdot 2 + 16^2 + 2$$

$$P(4^3) = 4P(4^2) + 16^3 + 2 = 4(4^3 + 4^1 \cdot 16^1 + 4 \cdot 2 + 16^2 + 2) + 16^3 + 2$$

$$= 4^4 + 4^2 \cdot 16^1 + 2 \cdot 4^2 + 4 \cdot 16^2 + 2 \cdot 4 + 16^3 + 2$$

$$= 4^4 + 4^2 \cdot 16^1 + 4 \cdot 16^2 + 16^3 + 2 \cdot 4^2 + 2 \cdot 4 + 2$$

$$= 4^4 + (4^2 \cdot 16^1 + 4 \cdot 16^2 + 16^3) + 2(4^2 + 4^1 + 4^0)$$

$$P(4^4) = 4P(4^3) + 16^4 + 2$$

$$= 4[4^4 + (4^2 \cdot 16^1 + 4^1 \cdot 16^2 + 16^3) + 2(4^2 + 4^1 + 4^0)] + 16^4 + 2$$

$$= 4^5 + (4^3 \cdot 16^1 + 4^2 \cdot 16^2 + 4^1 \cdot 16^3) + 2(4^3 + 4^2 + 4^1) + 4^0 \cdot 16^4 + 2 \cdot 4^0$$

$$= 4^5 + (4^3 \cdot 16^1 + 4^2 \cdot 16^2 + 4^1 \cdot 16^3 + 4^0 \cdot 16^4) + 2(4^3 + 4^2 + 4^1 + 4^0)$$

$$= 4^5 + 16(4^3 \cdot 16^0 + 4^2 \cdot 16^1 + 4 \cdot 16^2 + 4^0 \cdot 16^3) + 2(4^0 + 4^1 + 4^2 + 4^3)$$

⋮

$$\begin{aligned}
 P(4^m) = & \overbrace{4^{m+1}}^{\text{término generado por el caso base de la relación de recurrencia}} \\
 & \overbrace{+ 16(4^{m-1} \cdot 16^0 + 4^{m-2} \cdot 16^1 + \dots + 4^1 \cdot 16^{m-2} + 4^0 \cdot 16^{m-1})}^{\text{Sumatoria A}} \\
 & + \underbrace{2(4^0 + 4^1 + \dots + 4^{m-1})}_{\text{Sumatoria B}}
 \end{aligned}$$

Sumatoria A:

La Sumatoria A es una serie geométrica con primer término $a = 4^{m-1}$, razón constante $r = \frac{16}{4} = 4$ y potencia más grande igual a $m - 1$, reemplazando en la formula se tiene:

$$\begin{aligned}
 \frac{a \cdot r^{(\text{potencia más grande})+1} - a}{r - 1} &= \frac{4^{m-1}(4^{(m-1)+1}) - 4^{m-1}}{4 - 1} \\
 &= \frac{4^{m-1}(4^m) - 4^{m-1}}{3} \\
 &= 4^{m-1} \left(\frac{4^m - 1}{3} \right)
 \end{aligned}$$

Sumatoria B:

La Sumatoria B es una serie geométrica con primer término $a = 4^0 = 1$, razón constante $r = 4$ y potencia más grande igual a $m - 1$, reemplazando en la formula se tiene:

$$\begin{aligned}
 \frac{a \cdot r^{(\text{potencia más grande})+1} - a}{r - 1} &= \frac{1 \cdot (4^{(m-1)+1}) - 1}{4 - 1} \\
 &= \frac{4^m - 1}{3}
 \end{aligned}$$

Solución Relación de recurrencia:

$$\begin{aligned}
 P(4^m) = & (\text{término generado por el caso base de la relación de recurrencia}) \\
 & + 16 \cdot (\text{Sumatoria A}) + 2 \cdot (\text{Sumatoria B})
 \end{aligned}$$

$$\begin{aligned}
P(4^m) &= 4^{m+1} + 16 \left[4^{m-1} \left(\frac{4^m - 1}{3} \right) \right] + 2 \left[\frac{4^m - 1}{3} \right] \\
&= 4^{m+1} + 4^2 \cdot 4^{m-1} \left(\frac{4^m - 1}{3} \right) + 2 \left(\frac{4^m - 1}{3} \right) \\
&= 4^{m+1} + 4^{m+1} \left(\frac{4^m - 1}{3} \right) + 2 \left(\frac{4^m - 1}{3} \right) \\
&= 4^{m+1} + \left(4^{m+1} + 2 \right) \left(\frac{4^m - 1}{3} \right)
\end{aligned}$$

Se puede utilizar la técnica de demostración por inducción matemática para ratificar o refutar la validez de la solución de la relación de recurrencia.

4.2. Complejidad de algoritmos recursivos

Para poder calcular la complejidad de un algoritmo recursivo es necesario, primero, plantear una función de costo recursiva (también llamada relación de recurrencia) para contar la cantidad de operaciones que realiza el algoritmo recursivo, segundo, resolver la relación de recurrencia, y por último aplicar notaciones computacionales (ya sea O, Omega o Theta dependiendo del caso) sobre la solución de la relación de recurrencia para calcular la complejidad del algoritmo recursivo.

Ejemplo 9:

El siguiente algoritmo recursivo invierte el orden de los elementos almacenados en un vector A :

$Invertir(A[], i, j)$

1. *if* ($i < j$)
2. $u \leftarrow A[j]$
3. *for*($k = j; k > i; k--$)
4. $A[k] \leftarrow A[k - 1]$
5. $A[i] \leftarrow u$
6. *Invertir*($A[], i + 1, j$)

Cuando se hace el siguiente llamado a la función $Invertir(A[1, 2, 3], 1, 3)$ el resultado final del vector A cuando la recursividad termina es $[3, 2, 1]$.

Sea $T(n)$ la función recursiva (o relación de recurrencia) que cuenta la cantidad de operaciones que realiza el algoritmo *Invertir* para cuando este trabaja con un vector A de n elementos en el rango de i a j , por este motivo:

$$T(n) = \underbrace{\Theta(1)}_{\text{costo línea 1}} + \underbrace{\Theta(1)}_{\text{costo línea 2}} + \underbrace{\Theta(n)}_{\text{costo líneas 3 y 4}} + \underbrace{\Theta(1)}_{\text{costo línea 5}} + \underbrace{T(n-1)}_{\text{costo de Invertir } n-1 \text{ elementos}}$$

desde el punto de vista de notación theta se tiene que: $\Theta(1) + \Theta(1) + \Theta(n) + \Theta(1) = 3 \cdot \Theta(1) + \Theta(n) = \Theta(1) + \Theta(n) = \Theta(n)$, por este motivo

$$T(n) = \underbrace{\Theta(n)}_{\text{costo del algoritmo por llamado}} + \underbrace{T(n-1)}_{\text{costo de Invertir } n-1 \text{ elementos}}$$

$$T(n) = T(n-1) + \Theta(n), \text{ para } n \geq 2$$

Ahora hace falta definir el caso base de la relación de recurrencia, el cual se presenta cuando se quiere invertir un subvector que contiene un solo elemento, el orden inverso en este subvector es el mismo subvector y por este motivo no hay que realizar ningún trabajo, este caso es detectado en la línea 1 del algoritmo cuando $i = j$ que representa un subvector para trabajar de una sola posición. Cuando este caso se presenta el algoritmo recursivo termina su ejecución. El costo del caso base se representa como $T(1) = \Theta(1)$.

La relación de recurrencia completa es:

$$T(1) = \Theta(1)$$

$$T(n) = T(n-1) + \Theta(n), \text{ cuando } n \geq 2$$

de la relación de recurrencia anterior es removida la notación theta para que esta sea más fácil de resolver, de esta forma tenemos:

$$T(1) = 1$$

$$T(n) = T(n-1) + n, \text{ para } n \geq 2, n \in \mathbb{Z}^+$$

esta relación de recurrencia fue resuelta en el “Ejemplo 2” de éste capítulo, donde se obtuvo que la solución de la relación de recurrencia es:

$$T(n) = \frac{n \cdot (n+1)}{2}, \text{ para } n \geq 1, n \in \mathbb{Z}^+$$

en el “Ejemplo 32” de la sección 2.6 de “Notaciones computacionales”, se determinó que $1 + 2 + 3 + \dots + n = \frac{n \cdot (n+1)}{2}$ es $\Theta(n^2)$, por este motivo el algoritmo *Invertir* es de orden n^2 .

Ejemplo 10:

Sea el siguiente algoritmo recursivo donde las variables n , a , b y c son del tipo número natural:

```

PP(n, a, b, c)
1.  if (n > 0)
2.      PP(n-1, b, a, c)
3.      PP(n-1, c, b, a)
4.      printf(a, b)

```

Se quiere contar la cantidad de veces que se ejecuta la “línea 4” para un valor n .

La variable n es la única que se analiza para determinar si se realizan nuevos llamados recursivos a la función PP y si se imprime por pantalla los valores de las variables a y b . El valor más pequeño que puede tomar la variable n para que se realice este trabajo es uno, donde se realizará una impresión por pantalla (ejecución línea 4) y se harán dos llamados recursivos a la función PP con $n = 0$ que no generarán impresiones por pantalla, por este motivo el caso base de la relación de recurrencia es $T(1) = 1$. Ahora se define el caso recursivo de la relación de recurrencia, el cual se presenta cuando $n \geq 2$

$$\begin{aligned}
 T(n) &= \underbrace{T(n-1)}_{\text{impresiones realizadas por el primer llamado a la función } PP \text{ para } n-1} \\
 &+ \underbrace{T(n-1)}_{\text{impresiones realizadas por el segundo llamado a la función } PP \text{ para } n-1} \\
 &+ \underbrace{1}_{\text{ejecución línea 5}} \\
 T(n) &= \underbrace{2 \cdot T(n-1)}_{\text{impresiones realizadas por los dos llamados a la función } PP \text{ para } n-1} + \underbrace{1}_{\text{ejecución línea 5}}
 \end{aligned}$$

la relación de recurrencia completa es:

$$\begin{aligned}
 T(1) &= 1 \\
 T(n) &= 2 \cdot T(n-1) + 1, \text{ para } n \geq 2, n \in \mathbb{Z}^+
 \end{aligned}$$

Esta relación de recurrencia se resolvió en el “Ejemplo 4” de este capítulo, donde se obtuvo que $T(n) = 2^n - 1$, para $n \geq 1$, $n \in \mathbb{Z}^+$. Por lo tanto se realiza un total de $2^n - 1$ impresiones por pantalla (ejecución línea 4) para un valor de $n \geq 1$.

En la siguiente subsección se trabajará el análisis de la complejidad del algoritmo de ordenamiento por montones, donde se deja en evidencia que es fundamental entender muy bien el algoritmo sobre el cual se realiza el análisis de la complejidad, donde se hace uso de relaciones de recurrencia, sumatorias y notaciones computacionales (O , Ω y Θ).

4.2.1. El Algoritmo de Ordenamiento por Montones

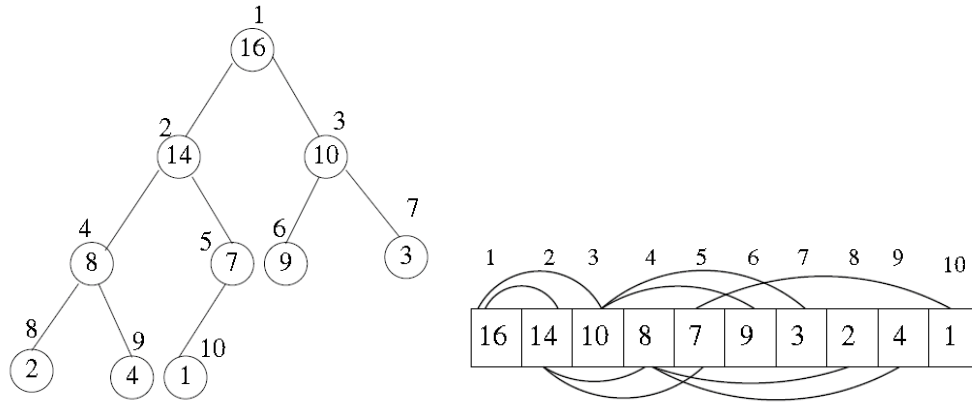
Toda esta subsección ha sido tomada de [M2010], donde dicha referencia tomo como punto de partida al libro de *Introducción a los Algoritmos* de Thomas Cormen y otros [CLRS2001].

El algoritmo de *ordenamiento por montones* creado por J. W. J. Williams en 1964 [W1964], hace uso de la estructura de datos montón para almacenar los números a ordenar. La estructura de datos montón también es utilizada para implementar de forma eficiente colas de prioridades. A continuación se presentan las propiedades de los

montones, la función que garantiza la propiedad de montón, la función para construir un montón y por último el algoritmo de ordenamiento por montones.

Los Montones y sus Propiedades

La estructura de datos montón es un arreglo de objetos muy parecido a un árbol binario completo. Cada nodo del árbol corresponde a un elemento del arreglo que almacena el valor en el nodo. El árbol está completamente lleno en todos los niveles excepto posiblemente en el nivel más bajo, el cual es llenado de izquierda a derecha hasta algún punto, la siguiente figura representa la situación anteriormente planteada con respecto a la forma que toma un árbol binario que representa un montón.



Un arreglo A que representa un montón es un objeto con dos atributos: $length[A]$, el cual es el número de elementos en el arreglo, y $heapSize[A]$, el cual es el número de elementos del montón almacenados en el arreglo A , donde $heapSize[A] \leq length[A]$.

El contenido de la raíz del árbol está almacenado en $A[1]$, y dado el índice de un nodo, el índice de su padre $Padre(i)$, el índice del hijo por la izquierda $Izq(i)$ y el índice del hijo por la derecha $Der(i)$ pueden ser calculados simplemente con las siguientes funciones:

$Padre(i)$
 $return \lfloor \frac{i}{2} \rfloor$

$Izq(i)$
 $return 2 \cdot i$

$Der(i)$
 $return 2 \cdot i + 1$

Hay dos clases de montones binarios: el *montón máximo* y el *montón mínimo*. En ambos casos, los valores de los nodos satisfacen una propiedad de montón, en el caso de un *montón máximo*, la propiedad del montón máximo es que para todo nodo i diferente de la raíz se cumple que $A[Padre(i)] \geq A[i]$, esto es que el valor de cada nodo como máximo es el valor de su padre, de esta forma el elemento más grande de un montón máximo

está almacenado en la raíz. En el caso de un *montón mínimo*, la propiedad del montón mínimo es que para todo nodo i diferente de la raíz se cumple que $A[Padre(i)] \leq A[i]$, esto es que el valor de cada nodo como mínimo es el valor de su padre, de esta forma el elemento más pequeño de un montón mínimo está almacenado en la raíz.

Al mirar un montón como un árbol, se define la *altura* de un nodo en el montón como la cantidad de aristas que hay desde el nodo a alguna de las hojas más lejanas que se alcanzan desde él en una ruta simple descendente, de esta forma la altura del montón es la altura del nodo raíz del árbol. La altura de un montón es $\theta(\lg n)$.

Garantizar la Propiedad de Montón

Los parámetros de entrada de la función *HeapifyMax* son un arreglo A y un subíndice i sobre el arreglo. La función *HeapifyMax* asume que tanto el subárbol izquierdo como el subárbol derecho a partir de la posición i son montones máximos, pero que $A[i]$ puede ser más pequeño que alguno de sus dos hijos, con lo que se violaría la propiedad de montón máximo. La función *HeapifyMax* le permite al valor almacenado en $A[i]$ “flotar hacia abajo” en el montón máximo hasta lograr que el subárbol con raíz i sea un montón máximo.

```

HeapifyMax( $A, i$ )
1.   $izq \leftarrow Izq(i)$ 
2.   $der \leftarrow Der(i)$ 
3.  if  $izq \leq heapSize[A]$  and  $A[izq] > A[i]$ 
4.      then  $posMax \leftarrow izq$ 
5.      else  $posMax \leftarrow i$ 
6.  if  $der \leq heapSize[A]$  and  $A[der] > A[posMax]$ 
7.      then  $posMax \leftarrow der$ 
8.  if  $posMax \neq i$ 
9.      then  $A[i] \leftrightarrow A[posMax]$ 
10.      $HeapifyMax(A, posMax)$ 

```

Si $A[i]$ es mayor o igual que la información almacenada en la raíz de los subárboles izquierdo y derecho entonces el árbol con raíz el nodo i es un montón máximo y la función termina. De lo contrario, la raíz de alguno de los subárboles tiene información mayor que la que se encuentra en $A[i]$ y es intercambiada con ésta, con lo cual se garantiza que el nodo i y sus hijos cumplen la propiedad de máximo montón, pero, sin embargo el subárbol hijo con el cual se intercambio la información de $A[i]$ ahora puede no cumplir la propiedad de máximo montón, por lo tanto, se debe llamar de forma recursiva a la función *HeapifyMax* sobre el subárbol hijo con el cual se hizo el intercambio.

La complejidad en el peor de los casos de la función *HeapifyMax* es $O(h)$, donde h es la altura del montón, como la altura del montón es $\theta(\lg n)$ entonces *HeapifyMax* es

$O(\lg n)$.

Construcción de un Montón

Se puede utilizar la función *HeapifyMax* de la manera abajo-arriba para convertir una arreglo A de n elementos en un montón máximo. Los elementos del arreglo almacenados en las posiciones $\lfloor \frac{n}{2} \rfloor + 1, \lfloor \frac{n}{2} \rfloor + 2, \dots, n$ son todos hojas del árbol, donde cada hoja es un montón de un solo elemento donde se cumple la propiedad de máximo montón.

BuildHeapMax(A)

1. $heapSize[A] \leftarrow length[A]$
2. for $i \leftarrow \lfloor \frac{length[A]}{2} \rfloor$ down to 1
3. do *HeapifyMax*(A, i)

En un *análisis sencillo* el peor de los casos de la función *BuildHeapMax* es $O(n \lg n)$, el cual se obtiene por los $O(n)$ llamados que se realizan de la función *HeapifyMax* que tiene complejidad $O(\lg n)$, donde $O(n) \cdot O(\lg n) = O(n \lg n)$.

En un *análisis más exacto* de la función si el montón tiene n nodos la altura del árbol binario es $h = \lfloor \lg n \rfloor$, como el costo computacional de la función *HeapifyMax* depende es de la altura del nodo sobre el que se esté trabajando, por este motivo en la siguiente tabla se presenta la cantidad de nodos con respecto a su altura en el árbol.

Altura del nodo	Cantidad de nodos
h	1
$h - 1$	2
$h - 2$	2^2
\vdots	\vdots
$h - (h - 1) = 1$	2^{h-1}
0	$\leq 2^h$

Adicionalmente se tiene la fórmula $\lceil \frac{n}{2^{h+1}} \rceil$ para calcular la cantidad máxima de nodos que se encuentra a una altura h en un montón que tiene n elementos. Teniendo en cuenta lo anterior, entonces, el análisis de la complejidad del peor de los casos de la función *BuildHeapMax* se puede realizar de la siguiente forma:

$$\sum_{h=1}^{\lfloor \lg n \rfloor} \left\lceil \frac{n}{2^{h+1}} \right\rceil O(h) = O\left(n \cdot \sum_{h=1}^{\lfloor \lg n \rfloor} \frac{h}{2^h}\right) = O\left(n \cdot \sum_{h=1}^{\infty} \frac{h}{2^h}\right)$$

se tiene la siguiente solución para la sumatoria ¹

$$\sum_{h=1}^m \frac{h}{2^h} = 2 - \frac{m+2}{2^m}, \quad m \in \mathbb{Z}^+$$

¹En el Ejemplo 23 del capítulo 2 se encuentra el desarrollo que permite determinar la solución de ésta sumatoria.

de donde se obtiene que

$$\sum_{h=1}^{\infty} \frac{h}{2^h} = 2$$

por lo tanto la complejidad de *BuildHeapMax* en el peor de los casos es:

$$\begin{aligned} O\left(n \cdot \sum_{h=1}^{\lceil \lg n \rceil} \frac{h}{2^h}\right) &= O\left(n \cdot \sum_{h=1}^{\infty} \frac{h}{2^h}\right) \\ &= O(n \cdot 2) \\ &= O(n) \end{aligned}$$

El Algoritmo de Ordenamiento por Montones

El algoritmo *HeapSort* inicialmente construye un *montón máximo* en el arreglo A de n posiciones, como el elemento más grande del arreglo se encuentra en $A[1]$ entonces este es intercambiado con el elemento que se encuentra almacenado en $A[n]$, de esta forma se tendrá ya ordenado el elemento más grande del arreglo en la posición que le corresponde, ahora toca decrementar el tamaño del montón y llamar a la función *HeapifyMax* para que se hagan los cambios (si es que son necesarios) para garantizar que en las primeras $n - 1$ posiciones del arreglo hay un montón. En el ciclo de repetición *for* de las líneas 2-5 se sigue intercambiando la información de la posición uno del arreglo con la que se encuentra en la última posición válida del montón, se sigue disminuyendo el tamaño del montón y se sigue llamando a la función *HeapifyMax* hasta tener un montón de tamaño uno. Cuando termina de ejecutarse el algoritmo el arreglo queda con su información ordenada de forma ascendente.

HeapSort(A)

1. *BuildHeapMax*(A)
2. *for* $i \leftarrow \text{length}[A]$ *down to* 2
3. do $A[1] \leftrightarrow A[i]$
4. $\text{heapSize}[A] \leftarrow \text{heapSize}[A] - 1$
5. *HeapifyMax*($A, 1$)

El algoritmo *HeapSort* tiene una complejidad de $O(n \lg n)$, complejidad que se genera en el ciclo de repetición *for* de las líneas 2-5, en donde el ciclo *for* ejecuta $n - 1$ veces el llamado de la función *HeapifyMax* que tiene una complejidad de $O(\lg n)$.

Ejercicios

1. Resolver las siguientes relaciones de recurrencia utilizando el *Método de Iteración*:

a.) $P(1) = \frac{1}{4}$

$$P(n) = P(n - 1) + n^3 - \frac{3}{4}n^2, \text{ para } n \geq 2, n \in \mathbb{Z}^+.$$

b.) $P(1) = \frac{1}{2}$
 $P(n) = P(n-1) + n^2 - \frac{1}{2}n$, para $n \geq 2$, $n \in \mathbb{Z}^+$.

c.) $P(1) = \frac{1}{5}$
 $P(n) = P(n-1) + n^3 - \frac{4}{5}n$, para $n \geq 2$, $n \in \mathbb{Z}^+$.

d.) $P(1) = \frac{3}{2}$
 $P(n) = P(n-1) + 2n - \frac{n}{2^n}$, para $n \geq 2$, $n \in \mathbb{Z}^+$.

e.) $P(1) = 1$
 $P(n) = 3P(\frac{n}{2}) + n$, para $n = 2^m$, $m \in \mathbb{Z}^+$.

f.) $P(1) = 1$
 $P(n) = 3P(\frac{n}{4}) + n$, para $n = 4^m$, $m \in \mathbb{Z}^+$.

g.) $P(1) = 1$
 $P(n) = 2P(\frac{n}{3}) + n$, para $n = 3^m$, $m \in \mathbb{Z}^+$.

h.) $P(1) = 1$
 $P(n) = 4P(\frac{n}{3}) + n$, para $n = 3^m$, $m \in \mathbb{Z}^+$.

i.) $P(1) = 1$
 $P(n) = 9P(\frac{n}{3}) + n$, para $n = 3^m$, $m \in \mathbb{Z}^+$.

2. Demostrar o refutar por inducción matemática que las relaciones de recurrencia siguientes tienen la solución que se plantea para cada una de ellas :

a.) $P(1) = 1$
 $P(n) = P(n-1) + 4n - 3$, para $n \geq 2$, $n \in \mathbb{Z}^+$.

Solución:

$$P(n) = 2n^2 - n, \text{ para } n \geq 1$$

b.) $P(1) = 2$
 $P(n) = P(n-1) + n^2 + n$, para $n \geq 2$, $n \in \mathbb{Z}^+$.

Solución:

$$P(n) = \frac{n(n+1)(n+2)}{3}, \text{ para } n \geq 1$$

c.) $P(1) = \frac{2}{3}$

$$P(n) = P(n-1) + n^2 - \frac{1}{3}n, \text{ para } n \geq 2, n \in \mathbb{Z}^+.$$

Solución:

$$P(n) = \frac{n^2(n+1)}{3}, \text{ para } n \geq 1$$

d.) $P(1) = 0$

$$P(n) = P(n-1) + n^2 - 1, \text{ para } n \geq 2, n \in \mathbb{Z}^+.$$

Solución:

$$P(n) = \frac{n(2n+5)(n-1)}{6}, \text{ para } n \geq 1$$

e.) $P(1) = 1$

$$P(n) = 3P(n-1) + 2^n - 1, \text{ para } n \geq 2, n \in \mathbb{Z}^+.$$

Solución:

$$P(n) = \frac{3^{n+1}+1}{2} - 2^{n+1}, \text{ para } n \geq 1$$

4.3. Preguntas tipo ECAES

1. Sea la siguiente relación de recurrencia²:

$$P(1) = 1$$

$$P(n) = 2P\left(\frac{n}{2}\right) + n, \text{ donde } n = 2^m, \text{ para } m \geq 1.$$

Determine el valor de $P(32)$

A.) 192

B.) 120

C.) 32

D.) 320

E.) 80

2. Sea la siguiente relación de recurrencia:

$$P(1) = 1$$

$$P(n) = 5P\left(\frac{n}{5}\right) + n, \text{ donde } n = 5^m \text{ y } m \geq 1.$$

Determine el valor de $P(125)$

A.) 25

B.) 125

C.) 250

D.) 500

E.) 1000

3. Sea la siguiente relación de recurrencia:

$$T(0) = 1$$

$$T(n) = T(n-1) + 2^n, \text{ para } n \geq 1$$

Determine el valor de $T(4)$

A.) 4

B.) 31

C.) 10

D.) 32

E.) 9

²Las preguntas desde la 1 hasta la 3 son del autor.

4. Sea la siguiente relación de recurrencia:

$$a_0 = 2,$$

$$a_1 = 6,$$

$$a_n = 6a_{n-1} - 9a_{n-2}, \text{ para } n \geq 2.$$

¿Cuál es el valor de a_4 ?

A.) 10

B.) 18

C.) 54

D.) 162

E.) 200

5. Una persona se encuentra ubicada en el tercer escalón de una escalera de n escalones y quiere saber la cantidad de posibilidades que existen para subir los $n - 3$ escalones restantes si puede subir de a uno o de dos escalones como quiera. ¿Cuál es la relación de recurrencia que sirve para contar todas estas posibilidades?

A.) $a_1 = 1,$

$$a_2 = 2,$$

$$a_n = a_{n-2} + a_{n-1}, \text{ para } n \geq 3.$$

B.) $a_1 = 1,$

$$a_2 = 1,$$

$$a_n = a_{n-2} + a_{n-1}, \text{ para } n \geq 3.$$

C.) $a_0 = 0,$

$$a_1 = 1,$$

$$a_n = a_{n-2} + a_{n-1}, \text{ para } n \geq 2.$$

D.) $a_1 = 1,$

$$a_2 = 1,$$

$$a_n = a_{n-2} + a_{n-1}, \text{ para } n \geq 2.$$

E.) $a_1 = 1,$

$$a_2 = 2,$$

$$a_n = 2a_{n-2} + a_{n-1}, \text{ para } n \geq 3.$$

6. Para un algoritmo recursivo, se ha planteado una ecuación de recurrencia de la forma

$$T(1) = \Theta(1)$$

$$T(n) = T(n-1) + \Theta(n), \text{ para } n > 1$$

El orden de complejidad de este algoritmo está dado por:

- A.) $\Theta(n^2)$
- B.) $\Theta(n^{\log_2 n})$
- C.) $\Theta(n)$
- D.) $\Theta(\log_2 n)$
- E.) $\Theta(2^{n-1})$

7. Para un algoritmo recursivo, se ha planteado una ecuación de recurrencia de la forma

$$T(1) = \Theta(1)$$

$$T(n) = T(n-1) + \Theta(2^n), \text{ para } n > 1$$

El orden de complejidad de este algoritmo está dado por

- A.) $\Theta(n^2)$
- B.) $\Theta(n^{\log_2 n})$
- C.) $\Theta(n)$
- D.) $\Theta(\log_2 n)$
- E.) $\Theta(2^n)$

8. Para un algoritmo recursivo, se ha planteado una ecuación de recurrencia de la forma

$$T(1) = \Theta(1)$$

$$T(n) = 2T\left(\frac{n}{2}\right) + \Theta(n), \text{ para } n > 1$$

El orden de complejidad de este algoritmo está dado por:

- A.) $\Theta(n^2)$
- B.) $\Theta(n^{\log_2 n})$
- C.) $\Theta(n)$
- D.) $\Theta(\log n)$
- E.) $\Theta(n \log n)$

9. Para un algoritmo recursivo, se ha planteado una ecuación de recurrencia de la forma

$$T(1) = \Theta(1)$$

$$T(n) = 2T(\frac{n}{2}) + \Theta(1), \text{ para } n > 1$$

El orden de complejidad de este algoritmo está dado por:

- A.) $\Theta(n^2)$
- B.) $\Theta(n^{\log_2 n})$
- C.) $\Theta(n)$
- D.) $\Theta(\log n)$
- E.) $\Theta(n \log n)$

10. Para un algoritmo recursivo, se ha planteado una ecuación de recurrencia de la forma

$$T(1) = \Theta(1)$$

$$T(n) = T(n-1) + \Theta(1), \text{ para } n > 1$$

El orden de complejidad de este algoritmo está dado por

- A.) $\Theta(n^2)$
- B.) $\Theta(n^{\log_2 n})$
- C.) $\Theta(n)$
- D.) $\Theta(\log_2 n)$
- E.) $\Theta(2^n)$

11. Una **precondición** es una condición que previamente se tiene que cumplir antes del llamado de una función para que ésta trabaje bien.

Sea la siguiente función

Incognita($A[], p, q, r$)

$$1. \quad n_1 = q - p + 1$$

$$2. \quad n_2 = r - q$$

Se crean los arreglos $L[1...n_1 + 1]$ y $R[1...n_2 + 1]$

$$3. \quad \text{for } i = 1 \text{ to } n_1$$

$$4. \quad \quad L[i] = A[p + i - 1]$$

$$5. \quad \text{for } j = 1 \text{ to } n_2$$

$$6. \quad \quad R[j] = A[q + j]$$

$$7. \quad L[n_1 + 1] = \infty$$

$$8. \quad R[n_2 + 1] = \infty$$

$$9. \quad i = 1$$

$$10. \quad j = 1$$

$$11. \quad \text{for } k = p \text{ to } r$$

```

12.    if ( $L[i] \leq R[j]$ )
13.         $A[k] = L[i]$ 
14.         $i = i + 1$ 
15.    else
16.         $A[k] = R[j]$ 
17.         $j = j + 1$ 

```

Cuál es el estado final del vector $A[]$ cuando se hace el siguiente llamado a la función

Incognita($A[2, 4, 1, 3], 1, 2, 4$)?

- A.) $A[2, 4, 1, 3]$
 - B.) $A[1, 3, 2, 4]$
 - C.) $A[4, 3, 2, 1]$
 - D.) $A[1, 2, 3, 4]$
 - E.) $A[1, 1, 1, 1]$
12. Qué hace la función *incognita* cuando se exige como precondition que los elementos del vector $A[]$ estén ordenados de forma ascendente desde la posición p hasta la q , así también, que los elementos estén ordenados de forma ascendente desde la posición $q + 1$ hasta la r ?
- A.) Los elementos quedan ordenados de forma ascendente en el vector $A[]$ desde la posición p hasta la r .
 - B.) Los elementos quedan ordenados de forma descendente en el vector $A[]$ desde la posición p hasta la r .
 - C.) En el vector $A[]$ quedan primero almacenados los elementos del vector $R[]$ y después los elementos del vector $L[]$.
 - D.) En el vector $A[]$ quedan primero almacenados los elementos del vector $L[]$ y después los elementos del vector $R[]$.
 - E.) Los elementos en el vector $A[]$ conservan sus posiciones originales.
- 13.Cuál es la complejidad de la función *incognita*?
- A.) $\Theta(p + r)$
 - B.) $\Theta(r - p)$
 - C.) $\Theta(p)$
 - D.) $\Theta(r)$
 - E.) $\Theta(2(p + r))$

14. Sea la siguiente función que hace uso de la función *Incognita* que ya se ha trabajado en puntos anteriores

```

Misterio( $A[]$ ,  $p$ ,  $r$ )
1. if ( $p < r$ )
2.    $q = \lfloor \frac{p+r}{2} \rfloor$ 
3.   Misterio( $A[]$ ,  $p$ ,  $q$ )
4.   Misterio( $A[]$ ,  $q+1$ ,  $r$ )
5.   Incognita( $A[]$ ,  $p$ ,  $q$ ,  $r$ )

```

Cuál es el estado final del vector $A[]$ cuando se hace el siguiente llamado a la función

$Misterio(A[4, 1, 3, 2], 1, 4)$?

- A.) $A[4, 1, 3, 2]$
 - B.) $A[3, 2, 4, 1]$
 - C.) $A[4, 3, 2, 1]$
 - D.) $A[1, 2, 3, 4]$
 - E.) $A[1, 1, 1, 1]$
15. Cuál es el efecto sobre el vector $A[]$ después de llamar a la función *Misterio*?
- A.) El vector $A[]$ queda como originalmente estaba.
 - B.) El vector $A[]$ queda ordenado de forma descendente.
 - C.) El vector $A[]$ queda ordenado de forma ascendente.
 - D.) Los números primos son los únicos que se conservan en el vector $A[]$.
 - E.) Primero quedan almacenados en el vector $A[]$ los números pares y luego los números impares.
16. Sea $n = r - p + 1$, la cantidad de elementos del vector $A[]$ sobre los cuales se va a trabajar en el llamado de la función *Misterio*. Para simplificar los cálculos se considera que n es una potencia de 2. Cuál es la relación de recurrencia que se genera a partir de la función *Misterio* para contar la cantidad de operaciones que ésta hace?
- A.) $M(1) = 1, M(n) = M(\frac{n}{2}) + n$
 - B.) $M(1) = 1, M(n) = 2M(n-1) + n$
 - C.) $M(1) = 1, M(n) = 2M(\frac{n}{2}) + 1$
 - D.) $M(1) = 1, M(n) = 2M(n-1) + 1$
 - E.) $M(1) = 1, M(n) = 2M(\frac{n}{2}) + n$
17. Cuál es la complejidad de la función *Misterio*?

- A.) $\Theta(\log n)$
- B.) $\Theta(n)$
- C.) $\Theta(n \log n)$
- D.) $\Theta(n^2)$
- E.) $\Theta(2^n)$

18. Sea el siguiente algoritmo recursivo:

```

Enigma( $A[ ]$ ,  $posicion$ )
1.  if ( $posicion > 1$ )
2.      for( $j = 1$ ;  $j \leq posicion - 1$ ;  $j++$ )
3.          if ( $A[j] > A[j + 1]$ )
4.               $A[j] \leftrightarrow A[j + 1]$ 
5.          Enigma( $A[ ]$ ,  $posicion - 1$ )

```

Cuál es el estado final del vector $A[]$ cuando se hace el siguiente llamado a la función $Enigma(A[4, 2, 1, 3], 4)$?

- A.) $A[4, 3, 2, 1]$
- B.) $A[3, 4, 1, 2]$
- C.) $A[4, 2, 1, 3]$
- D.) $A[3, 1, 2, 4]$
- E.) $A[1, 2, 3, 4]$

19. ¿Cuál de las siguientes es la relación de recurrencia que sirve para contar la cantidad de operaciones que hace la función $Enigma$ cuando se llama con un vector de n elementos?

- A.) $T(1) = \Theta(1)$
 $T(n) = T(n - 1) + \Theta(1)$, cuando $n > 1$
- B.) $T(1) = \Theta(1)$
 $T(n) = T(n - 1) + \Theta(n)$, cuando $n > 1$
- C.) $T(1) = \Theta(1)$
 $T(n) = 2T(\frac{n}{2}) + \Theta(n)$, cuando $n > 1$
- D.) $T(1) = \Theta(1)$
 $T(n) = T(n - 1) + \Theta(n^2)$, cuando $n > 1$
- E.) $T(1) = 1$
 $T(n) = T(n - 1) + 2n$, cuando $n > 1$

20. ¿Cuál es la complejidad de la función $Enigma$ en términos de la notación Θ ?

- A.) $\Theta(\sqrt{n})$

- B.) $\Theta(n)$
 C.) $\Theta(n \lg n)$
 D.) $\Theta(n^2)$
 E.) $\Theta(n^3)$
21. María escoge un número entre 1 y 64. Pedro debe identificar el número haciendo preguntas que se responden con un “sí” o con un “no”. Pedro sabe que María siempre responde con la verdad. Si Pedro usa una estrategia óptima ¿cuántas preguntas debe hacer en el peor de los casos?
- A.) 1
 B.) 32
 C.) 6
 D.) 5
 E.) 7
22. Sea el siguiente algoritmo recursivo:

```

BB(A[ ], i, j, n )
1.  if (i > j)
2.      return -1
3.  else
4.      m ← ⌊  $\frac{i+j}{2}$  ⌋
5.      if (A[m] == n)
6.          return m
7.      else
8.          if (n > A[m])
9.              BB(A[ ], m + 1, j, n )
10.         else
11.             BB(A[ ], i, m - 1, n )

```

¿Cuál es el resultado que se obtiene cuando se hace el siguiente llamado a la función $BB(A[2, 4, 7, 9, 10], 1, 5, 8)$?

- A.) -1
 B.) 1
 C.) 2
 D.) 4
 E.) 5
23. ¿Cuál de las siguientes es la relación de recurrencia que mejor representa la cantidad de operaciones que hace la función BB (ejercicio 22) cuando se llama con un vector de n elementos?

- A.) $T(1) = 1$
 $T(n) = T(n - 1) + n$, cuando $n > 1$
- B.) $T(1) = 1$
 $T(n) = T(\lfloor \frac{n}{2} \rfloor) + n$, cuando $n > 1$
- C.) $T(1) = 1$
 $T(n) = T(\lfloor \frac{n}{2} \rfloor) + 1$, cuando $n > 1$
- D.) $T(1) = 1$
 $T(n) = 2T(\lfloor \frac{n}{2} \rfloor) + 1$, cuando $n > 1$
- E.) $T(1) = 1$
 $T(n) = 2T(\lfloor \frac{n}{2} \rfloor) + n$, cuando $n > 1$
24. ¿Cuál es la complejidad de la función *BB* (ejercicio 22) en términos de la notación *O* para el peor de los casos cuando la función es llamada con un vector de n elementos?
- A.) $O(n^2)$
- B.) $O(1)$
- C.) $O(\log n)$
- D.) $O(n)$
- E.) $O(n \log n)$
25. Sea el siguiente algoritmo recursivo:

```

PP(n, a, b, c)
1.  if (n > 0)
2.      {
3.          PP(n - 1, b, a, c)
4.          PP(n - 1, c, b, a)
5.          printf(a, b)
6.      }

```

Cuál es el resultado total de las impresiones que son generadas por los llamados recursivos a la función cuándo se hace el siguiente llamado inicial a la función $PP(2, 1, 2, 3)$?

- A.) 2, 1, 3, 2, 2, 1
- B.) 1, 2, 2, 1, 3, 2
- C.) 3, 2, 2, 1, 1, 2
- D.) 2, 1, 3, 2, 1, 2
- E.) 1, 2, 1, 2, 1, 2

Capítulo 5

Conjuntos

Definición:

Un conjunto es una colección de objetos en la que no importa el orden en que estos aparecen.

Hay varias formas para describir un conjunto. Una forma es describir todos los elementos del conjunto, cuando esto es posible, entre llaves separados por comas.

Ejemplo 1:

- El conjunto de todas las vocales del alfabeto es: $V = \{a, e, i, o, u\}$.
- el conjunto de todos los enteros impares positivos menores que 10 es $I = \{1, 3, 5, 7, 9\}$.

Otra forma para describir un conjunto sin listar todos los elementos del conjunto es, listar algunos elementos del conjunto y colocar puntos suspensivos cuando el patrón general de los elementos es obvio.

Ejemplo 2:

El conjunto de los enteros positivos estrictamente menores que 100 es $E = \{1, 2, 3, \dots, 99\}$

Algunos de los conjuntos más importantes en matemáticas discretas son:

- $\mathbb{N} = \{0, 1, 2, 3, \dots\}$ El conjunto de los números naturales
- $\mathbb{Z} = \{\dots, -3, -2, -1, 0, 1, 2, 3, \dots\}$ El conjunto de los números enteros
- $\mathbb{Z}^+ = \{1, 2, 3, 4, \dots\}$ El conjunto de los números enteros positivos
- $\mathbb{Q} = \{\frac{p}{q} \mid p \in \mathbb{Z}, q \in \mathbb{Z}^+, q \neq 0\}$ El conjunto de los números racionales
- \mathbb{R} ; es el conjunto de los números reales.

Definición:

Dos conjuntos son iguales si y únicamente si tienen los mismos elementos.

Ejemplo 3:

Los conjuntos $A = \{1, 3, 5\}$ y $B = \{3, 5, 1\}$ son iguales porque contienen los mismos elementos.

Se debe tener en cuenta que el orden en que son presentados los elementos de un conjunto no tiene importancia, tampoco tiene importancia listar varias veces el mismo elemento de un conjunto, de todas maneras sigue siendo el mismo conjunto, por ejemplo el conjuntos $A = \{3, 3, 5, 3, 1, 3, 5\}$ es igual al conjunto $B = \{1, 3, 5\}$.

Otra forma para construir conjuntos es usando una “notación para construir conjuntos”. Todos los elementos del conjunto se caracterizan porque cumplen la propiedad de pertenencia al conjunto.

Ejemplo 4:

$$I = \{x \mid x \text{ es un número entero positivo impar menor que } 100\}$$

Los conjuntos pueden ser representados gráficamente utilizando diagramas de Venn. En los diagramas de Venn el conjunto universal U , el cual contiene todos los objetos bajo consideración, es representado por un rectángulo. Internamente en el rectángulo, los círculos u otra figura geométrica son utilizados para representar conjuntos.

Ejemplo 5:

Dibujar un diagrama de Venn para representar el conjunto de las vocales del alfabeto español.

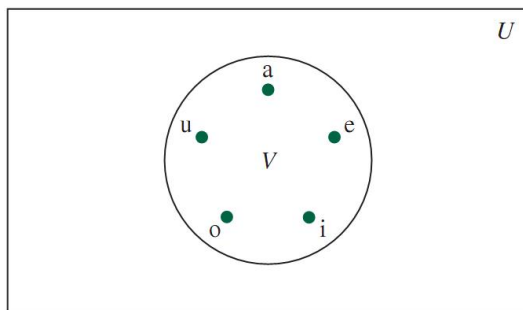


Diagrama de Venn para el conjunto de las vocales.

La pertenencia de un elemento a un conjunto se representa por $a \in A$. La notación $a \notin A$ indica que el elemento a no pertenece al conjunto A .

Tener en cuenta que las letras mayúsculas se utilizan para representar conjuntos y que las letras minúsculas se utilizan para representar elementos.

El conjunto vacío o nulo es aquel que no tiene elementos y se representa por $A = \{ \}$ o $A = \emptyset$.

Definición:

El conjunto A es subconjunto de B si y únicamente si cada elemento de A es también un elemento de B . La notación $A \subseteq B$ indica que A es un subconjunto del conjunto B .

Se puede decir que $A \subseteq B$ si y únicamente si $\forall_x (x \in A \rightarrow x \in B)$.

Ejemplo 6:

Sean los conjuntos: $B = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}$ $A = \{1, 2, 5, 7\}$, entonces $A \subseteq B$.

Teorema:

Para cualquier conjunto S , $\emptyset \subseteq S$ y $S \subseteq S$.

Cuando el conjunto A es un subconjunto de B pero $A \neq B$, entonces se escribe $A \subset B$ y se dice que A es un subconjunto propio de B .

Gráficamente un subconjunto propio se representa de la siguiente manera:

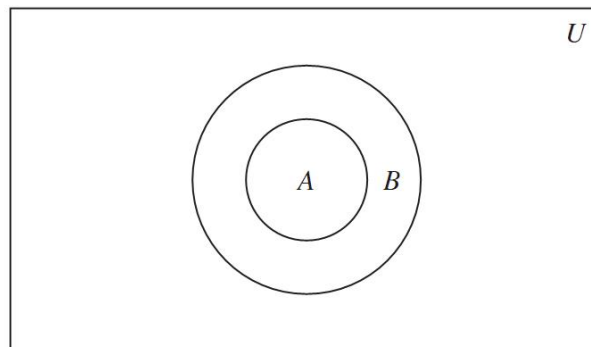


Diagrama de Venn para mostrar que A es un subconjunto propio de B .

Definición:

Sea S un conjunto. Si hay exactamente n elementos distintos en S donde n es un número natural, se dice que S es un conjunto finito y que n es la cardinalidad de S . La cardinalidad de S es denotada por $|S|$.

Ejemplo 7:

Sea el conjunto $A = \{1, 3, 5, 7, 3, 5, 1, 7\}$ $|A| = 4$.

Ejemplo 8:

$|\emptyset| = 0$, porque el conjunto vacío no tiene elementos.

5.1. El conjunto potencia**Definición:**

Dado un conjunto S , el conjunto potencia de S es el conjunto de todos los subconjuntos de S . El conjunto potencia de S es denotado por $\mathcal{P}(S)$.

Ejemplo 9:

¿Cuál es el conjunto potencia del conjunto $A = \{0, 1, 2\}$?

$$\mathcal{P}(A) = \{\{\}, \{0\}, \{1\}, \{2\}, \{0, 1\}, \{0, 2\}, \{1, 2\}, \{0, 1, 2\}\}$$

Si un conjunto A tiene cardinalidad n ($|A| = n$), entonces el conjunto potencia de A tiene 2^n elementos.

Ejemplo 10:

¿Cuántos elementos tiene $\mathcal{P}(A)$ si $|A| = 3$?

$$|\mathcal{P}(A)| = 2^{|A|} = 2^3 = 8$$

5.2. Producto cartesiano**Definición:**

La n -tupla ordenada $(a_1, a_2, a_3, \dots, a_n)$ es la colección ordenada que tiene a_1 como el primer elemento, a_2 como el segundo elemento, \dots , y a_n como el n -ésimo elemento.

Dos n -tuplas son iguales si y únicamente si cada par de sus elementos es igual. En otras palabras, $(a_1, a_2, a_3, \dots, a_n) = (b_1, b_2, b_3, \dots, b_n)$ si y únicamente si $a_i = b_i$, para $i = 1, 2, \dots, n$. En particular, una 2-tupla es llamada *par ordenado*.

Los pares ordenados (a, b) y (c, d) son iguales si y únicamente si $a = c$ y $b = d$. (a, b) y (b, a) no son iguales, a no ser que $a = b$.

Definición:

Sea A y B conjuntos. El producto cartesiano de A y B , denotado por $A \times B$, es el conjunto de todos los pares ordenados (a, b) donde $a \in A$ y $b \in B$. Donde $A \times B = \{(a, b) \mid a \in A \wedge b \in B\}$.

Ejemplo 11:

Sean los conjuntos $A = \{1, 2, 3, 4\}$ y $B = \{a, b, c\}$

- ¿Cuál es el conjunto $A \times B$?
- ¿Cuál es el conjunto $B \times A$?
- ¿Cuál es el conjunto $B \times B$?

$$A \times B = \{(1, a), (1, b), (1, c), \\ (2, a), (2, b), (2, c), \\ (3, a), (3, b), (3, c), \\ (4, a), (4, b), (4, c)\}$$

$$B \times A = \{(a, 1), (a, 2), (a, 3), (a, 4), \\ (b, 1), (b, 2), (b, 3), (b, 4), \\ (c, 1), (c, 2), (c, 3), (c, 4)\}$$

$$B \times B = \{(a, a), (a, b), (a, c), \\ (b, a), (b, b), (b, c), \\ (c, a), (c, b), (c, c)\}$$

Un subconjunto R del producto cartesiano $A \times B$ es llamado una relación del conjunto A al conjunto B . Los elementos de R son pares ordenados donde el primer elemento pertenece a A y el segundo pertenece a B .

Ejemplo 12:

Sea $A = \{1, 2, 3, 4\}$ y $B = \{1, 4, 9, 16\}$

$$A \times B = \{(1, 1), (1, 4), (1, 9), (1, 16), \\ (2, 1), (2, 4), (2, 9), (2, 16), \\ (3, 1), (3, 4), (3, 9), (3, 16), \\ (4, 1), (4, 4), (4, 9), (4, 16)\}$$

y sea $R = \{(1, 1), (2, 4), (3, 9), (4, 16)\}$ donde la segunda componente de cada pareja es el cuadrado de la primera.

Definición:

El producto cartesiano de los conjuntos A_1, A_2, \dots, A_n denotado por $A_1 \times A_2 \times \dots \times A_n$, es el conjunto ordenado de n -tuplas (a_1, a_2, \dots, a_n) donde $a_i \in A_i$ para $i = 1, 2, \dots, n$.

En otras palabras, $A_1 \times A_2 \times \dots \times A_n = \{(a_1, a_2, \dots, a_n) \mid a_i \in A_i \text{ para } i = 1, 2, \dots, n\}$.

Ejemplo 13:

Sea $A = \{1, 2\}$, $B = \{a, e\}$ y $C = \{x, y\}$, ¿cuál es el resultado de $A \times B \times C$?

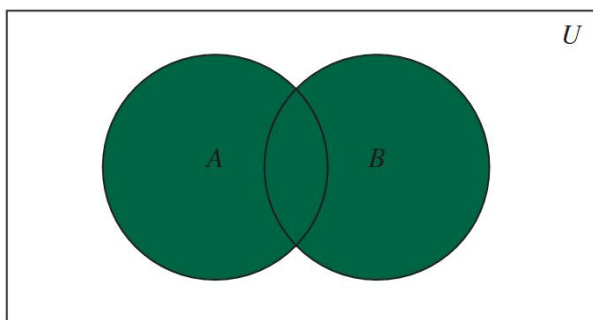
$$A \times B \times C = \{(1, a, x), (1, a, y), (1, e, x), (1, e, y), (2, a, x), (2, a, y), (2, e, x), (2, e, y)\}$$

5.3. Operaciones de conjuntos**Definición de la operación de unión:**

Sean A y B conjuntos. La unión de los conjuntos A y B denotada por $A \cup B$ es el conjunto que contiene elementos que son de A o B , o de ambos.

Un elemento x pertenece a la unión de A y B si y únicamente si x pertenece a A o x pertenece a B . Formalmente se tiene: $A \cup B = \{x \mid x \in A \vee x \in B\}$.

Gráficamente se tiene:



La parte sombreada es $A \cup B$.

Ejemplo 14:

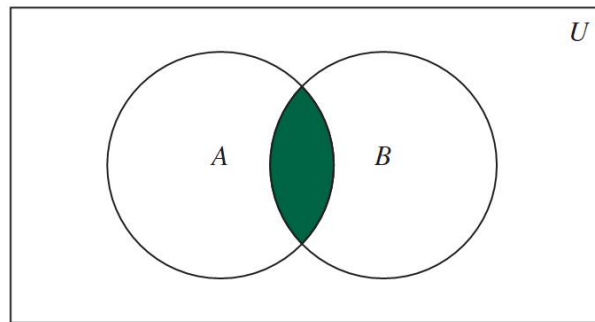
La unión de los conjuntos $A = \{1, 3, 5\}$ y $B = \{1, 2, 3\}$ es $A \cup B = \{1, 2, 3, 5\}$.

Definición de la operación de intersección:

Sean A y B conjuntos. La intersección de los conjuntos A y B denotada por $A \cap B$ es el conjunto que contiene los elementos que están al mismo tiempo en A y B .

Un elemento x pertenece a la intersección de los conjuntos A y B si y únicamente si x pertenece a A y x pertenece a B . Formalmente se tiene: $A \cap B = \{x \mid x \in A \wedge x \in B\}$.

Gráficamente se tiene:



La parte sombreada es $A \cap B$.

Ejemplo 15:

La intersección de los conjuntos $A = \{1, 3, 5\}$ y $B = \{1, 2, 3\}$ es $A \cap B = \{1, 3\}$.

Definición de conjuntos disyuntos:

Dos conjuntos A y B son disyuntos si su intersección ($A \cap B$) es vacía.

Ejemplo 16:

Sea $A = \{1, 3, 5, 7, 9\}$ y $B = \{2, 4, 6, 8, 10\}$. $A \cap B = \{ \}$, por lo tanto A y B son disyuntos.

La cardinalidad de la unión de los conjuntos A y B es:

$$|A \cup B| = |A| + |B| - |A \cap B|.$$

La generalización de la cardinalidad para la unión de un número arbitrario de conjuntos es llamado el principio de inclusión-exclusión.

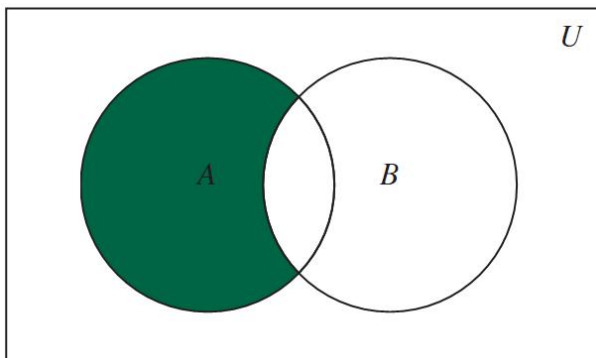
Definición de la operación de diferencia:

Sean A y B conjuntos. La diferencia de los conjuntos A y B denotada por $A - B$, es el conjunto que contiene los elementos que están en A pero no en B . La diferencia de A

y B es también llamada el complemento de B con respecto a A .

Un elemento x pertenece a la diferencia de A y B si y únicamente si $x \in A$ y $x \notin B$. Formalmente se tiene: $A - B = \{x \mid x \in A \wedge x \notin B\}$.

Gráficamente se tiene:



La parte sombreada es $A - B$.

Ejemplo 17:

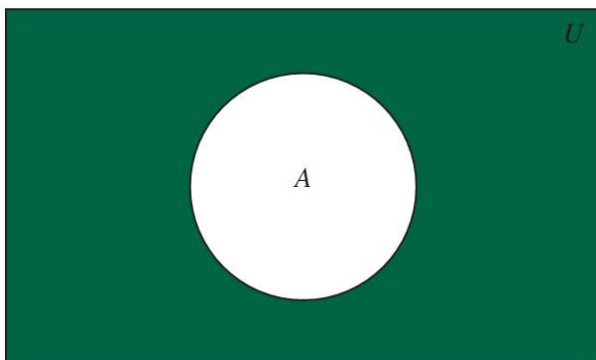
La diferencia de los conjuntos $A = \{1, 3, 5\}$ y $B = \{1, 2, 3\}$ es $A - B = \{5\}$, $B - A = \{2\}$.

Definición de la operación de complemento:

Sea \mathcal{U} el conjunto universo, el complemento del conjunto A , denotado por \bar{A} , es el complemento del conjunto A con respecto a \mathcal{U} . En otras palabras, el complemento del conjunto A es $\mathcal{U} - A$.

Un elemento x pertenece a \bar{A} si y únicamente si $x \notin A$. Formalmente se tiene: $\bar{A} = \{x \mid x \notin A\}$.

Gráficamente se tiene:



La parte sombreada es \bar{A} .

Ejemplo 18:

Sea $A = \{a, e, i, o, u\}$ y el universo son todas las letras del alfabeto español. Entonces $\overline{A} = \{b, c, d, f, g, h, j, k, l, m, n, \tilde{n}, p, q, r, s, t, v, w, x, y, z\}$.

5.4. Identidades en conjuntos

En la siguiente tabla las letras mayúsculas A , B y C representan conjuntos, el conjunto universal es representado con el símbolo \mathcal{U} y el conjunto vacío es representado con el símbolo \emptyset .

Identidad	Nombre
$A \cup \emptyset = A$	Ley de Identidad
$A \cap \mathcal{U} = A$	Ley de Identidad
$A \cup \mathcal{U} = \mathcal{U}$	Ley de Dominación
$A \cap \emptyset = \emptyset$	Ley de Dominación
$A \cup A = A$	Ley de Idempotencia
$A \cap A = A$	Ley de Idempotencia
$\overline{\overline{A}} = A$	Ley de Doble Complemento
$A \cup B = B \cup A$	Ley Conmutativa
$A \cap B = B \cap A$	Ley Conmutativa
$A \cup (B \cap C) = (A \cup B) \cap C$	Ley Asociativa
$A \cap (B \cup C) = (A \cap B) \cup (A \cap C)$	Ley Asociativa
$A \cap (B \cup C) = (A \cap B) \cup (A \cap C)$	Ley Distributiva
$A \cup (B \cap C) = (A \cup B) \cap (A \cup C)$	Ley Distributiva
$\overline{A \cup B} = \overline{A} \cap \overline{B}$	Ley de De Morgan
$\overline{A \cap B} = \overline{A} \cup \overline{B}$	Ley de De Morgan
$A \cup (A \cap B) = A$	Ley de Absorción
$A \cap (A \cup B) = A$	Ley de Absorción
$A \cup \overline{A} = \mathcal{U}$	Ley de Complemento
$A \cap \overline{A} = \emptyset$	Ley de Complemento

Ejemplo 19:

Probar que $\overline{A \cap B} = \overline{A} \cup \overline{B}$.

Estrategia: Se debe probar que los dos conjuntos son iguales al mostrar que cada uno es subconjunto del otro.

- $\overline{A \cap B} \subseteq \overline{A} \cup \overline{B}$?.

$x \in \overline{A \cap B}$, $x \notin (A \cap B)$, $\neg(x \in A \wedge x \in B)$, por la ley de Morgan
 $\neg(x \in A) \vee \neg(x \in B)$, $x \notin A \vee x \notin B$, $x \in \overline{A} \vee x \in \overline{B}$ por la
 definición de complemento, $x \in (\overline{A} \cup \overline{B})$ por la definición de la unión, por lo
 tanto $\overline{A \cap B} \subseteq \overline{A} \cup \overline{B}$.

- $\overline{A} \cup \overline{B} \subseteq \overline{A \cap B}$?.

$x \in (\overline{A} \cup \overline{B})$,
 $x \in \overline{A} \vee x \in \overline{B}$ definición de la unión,
 $x \notin A \vee x \notin B$ definición de complemento,
 $\neg(x \in A) \vee \neg(x \in B)$ equivalencia lógica,
 $\neg((x \in A) \wedge (x \in B))$ ley de Morgan,
 $\neg(x \in (A \cap B))$ definición de la intersección,
 $x \notin (A \cap B)$, $x \in \overline{(A \cap B)}$ definición de complemento,
 por lo tanto $\overline{A} \cup \overline{B} \subseteq \overline{A \cap B}$.

Como cada lado es subconjunto del otro entonces queda demostrado que
 $\overline{A \cap B} = \overline{A} \cup \overline{B}$.

Ejemplo 20:

Usar notación y equivalencia lógica para probar que $\overline{A \cap B} = \overline{A} \cup \overline{B}$.

Estrategia: Utilizar una cadena de equivalencias para proveer una demostración de esta identidad.

$$\begin{aligned}
 \overline{A \cap B} &= \{x \mid x \notin (A \cap B)\} \\
 &= \{x \mid \neg(x \in (A \cap B))\} \\
 &= \{x \mid \neg(x \in A \wedge x \in B)\} \\
 &= \{x \mid \neg(x \in A) \vee \neg(x \in B)\} \\
 &= \{x \mid x \notin A \vee x \notin B\} \\
 &= \{x \mid x \in \overline{A} \vee x \in \overline{B}\} \\
 &= \{x \mid x \in (\overline{A} \cup \overline{B})\} \\
 &= \overline{A} \cup \overline{B}
 \end{aligned}$$

5.5. Uniones e intersecciones generalizadas

Definición de unión generalizada:

La unión de una colección de conjuntos es el conjunto que contiene todos los elementos que son miembros de al menos un conjunto de la colección. Se usa la notación:

$$A_1 \cup A_2 \cup A_3 \cup \dots \cup A_n = \bigcup_{i=1}^n A_i$$

Definición de intersección generalizada:

La intersección de una colección de conjuntos es el conjunto que contiene los elementos que son miembros de todos los conjuntos de la colección. Se usa la notación:

$$A_1 \cap A_2 \cap A_3 \cap \dots \cap A_n = \bigcap_{i=1}^n A_i$$

Ejemplo 21:

Sea $A_i = \{i, i+1, i+2, \dots\}$ donde

$$\bigcup_{i=1}^n A_i = \bigcup_{i=1}^n \{i, i+1, i+2, \dots\} = \{1, 2, 3, \dots\}$$

y

$$\bigcap_{i=1}^n A_i = \bigcap_{i=1}^n \{i, i+1, i+2, \dots\} = \{n, n+1, n+2, \dots\}$$

5.6. Ejercicios

1. Suponer que A es el conjunto de estudiantes de segundo semestre de Ingeniería de Sistemas de Jornada Especial de la Universidad Tecnológica y B el conjunto de estudiantes de Ingeniería de Sistemas de Jornada Especial que tienen matriculado el curso de Matemáticas II. Expresar estos conjuntos en términos de A y B .
 - a.) El conjunto de estudiantes de segundo semestre de Ingeniería de Sistemas de Jornada Especial de la Universidad Tecnológica que tienen matriculado el curso de Matemáticas II.
 - b.) El conjunto de estudiantes de segundo semestre de Ingeniería de Sistemas de Jornada Especial de la Universidad Tecnológica que no tienen matriculado el curso de Matemáticas II.
 - c.) El conjunto de estudiantes o bien que son segundo semestre de Ingeniería de Sistemas de Jornada Especial de la Universidad Tecnológica o bien que están matriculados en el curso de Matemáticas II.

- d.) El conjunto de estudiantes o bien que no son de segundo semestre de Ingeniería de Sistemas de Jornada Especial de la Universidad Tecnológica o bien que no están matriculados en el curso de Matemáticas II.
2. Un profesor tiene 24 libros de texto introductorios de *Ciencias de la Computación*, y le interesa saber en qué medida tratan los temas: (A) análisis de algoritmos, (B) estructuras de datos y (C) compiladores. Los siguientes datos presentan el número de libros que contienen material sobre estos temas:
- $|A| = 8$
 - $|B| = 13$
 - $|C| = 13$
 - $|A \cap B| = 5$
 - $|A \cap C| = 3$
 - $|B \cap C| = 6$
 - $|A \cap B \cap C| = 2$
- a.) ¿Cuántos textos incluyen material de exactamente uno de los temas?
- b.) ¿Cuántos textos incluyen material de exactamente dos de los temas?
- c.) ¿Cuántos textos tratan al menos alguno de los temas?
- d.) ¿Cuántos textos no tratan ninguno de los temas?
- e.) ¿Cuántos textos no tratan el tema de compiladores?
- f.) ¿Cuántos textos no tratan ni el tema de compiladores ni el tema de estructuras de datos?
3. Sean A y B conjuntos. Mostrar gráficamente utilizando diagramas de Venn que se cumplen cada una de las siguientes igualdades:
- a.) $(A \cap B) \cup (A \cap \overline{B}) = A$.
- b.) $A - B = A \cap \overline{B}$.
- c.) $A \cap (A \cup B) = A$.
- d.) $A \cup (A \cap B) = A$.
4. Sean A y B conjuntos. Simplificar las siguientes expresiones en la teoría de conjuntos, en cada paso indicar la ley utilizada. Validar el resultado obtenido utilizando diagramas de Venn. ¿El gráfico de la simplificación es igual al gráfico de la expresión original?.
- a.) $\overline{A} \cup (A \cap B) \cup \overline{B}$.
- b.) $\overline{A} \cap (A \cup B) \cap \overline{B}$.
- c.) $((\overline{A} \cap B) \cup (A \cap \overline{B})) \cap A$.

- d.) $((\overline{A} \cap B) \cup (A \cap \overline{B})) \cap \overline{A}$.
- e.) $((\overline{A} \cap B) \cup (A \cap \overline{B})) \cap \overline{B}$.
- f.) $((\overline{A} \cap B) \cup (A \cap \overline{B})) \cap B$.
5. Sean los conjuntos $A = \{1, 2, 3, 4\}$, $B = \{a, b, c\}$ y $C = \{a, 1, b, 2, c, 4\}$. ¿Cuál es el resultado de $(C \times C) - (B \times A)$?
6. Sean los conjuntos $A = \{1, 2, 3\}$, $B = \{a, b, c\}$ y $C = \{a, 1, b, 2\}$. ¿Cuál es el resultado de $(C \times C) - (B \times A)$?
7. Sean los conjuntos $A = \{1, 2, 3\}$, $B = \{a, b, c\}$ y $C = \{a, 1, b, 2\}$. ¿Cuál es el resultado de $(C \times C) - (A \times B)$?
8. Sean los conjuntos $A = \{1, 2, 3\}$, $B = \{a, b, c\}$ y $C = \{a, 1, b, 2, c\}$. ¿Cuál es el resultado de $(C \times C) - (B \times A)$?
9. Sean los conjuntos $A = \{1, 2, 3\}$, $B = \{a, b, c\}$ y $C = \{a, 1, b, 2, c\}$. ¿Cuál es el resultado de $(A \times B) - (C \times C)$?
10. Sea $A_i = \{i^i, i^{2i}, i^{3i}, \dots, i^{i*i}\}$. ¿Cuántos elementos tiene el conjunto resultado de $\cup_{i=1}^n (A_i)$?
11. Sea $A_i = \{2^i, 2^{i+1}, 2^{i+2}, \dots, 2^{2i}\}$. ¿Cuál es el conjunto resultado de $\cup_{i=0}^n A_i$?
12. Sea $A_i = \{2^i, 2^{i+1}, 2^{i+2}, \dots, 2^{2i}\}$. ¿Cuál es el conjunto resultado de $\cap_{i=2}^n A_i$?

5.7. Preguntas tipo ECAES

1. Sea $A_i = \{2^i, 2^{i+1}, 2^{i+2}, \dots, 2^{2i}\}$. ¿Cuál es el conjunto resultado de¹ $\cup_{i=2}^n A_i$?
 - A.) $\{2^n, 2^{n+1}, 2^{n+2}, \dots, 2^{2n}\}$
 - B.) $\{2^0, 2^1, 2^2, \dots, 2^{n+2}\}$
 - C.) $\{2^0, 2^1, 2^2, \dots, 2^{2n}\}$
 - D.) $\{2^2, 2^3, 2^4, \dots, 2^{2n}\}$
 - E.) $\{\}$
2. Sean A y B conjuntos cualesquiera. El conjunto potencia de la unión de los conjuntos A y B denotado por $P(A \cup B)$ es² :
 - A.) igual $P(A) \cup P(B)$
 - B.) subconjunto $P(A) \cup P(B)$
 - C.) contiene a $P(A) \cup P(B)$
 - D.) igual $P(A) \cup P(B) - \{\emptyset\}$
3. En una fiesta que cuenta con un total de 40 asistentes se tiene que 25 de ellos hablan inglés, 20 de ellos hablan frances y que 15 de ellos hablan tanto inglés como frances. ¿Cuántos de los asistentes a la fiesta no hablan ni inglés ni frances?³
 - A.) 25
 - B.) 20
 - C.) 15
 - D.) 10
 - E.) 5
4. En una encuesta a 300 estudiantes se encontró que 68 se comportan bien, 138 son inteligentes, 160 son habladores, 120 son habladores e inteligentes, 20 se comportan bien y son inteligentes, 18 se comportan bien y son habladores, 15 se comportan bien, son habladores y son inteligentes. ¿Cuántos de los estudiantes entrevistados no se comportan bien, no son habladores y no son inteligentes?
 - A.) 0
 - B.) 77
 - C.) 223

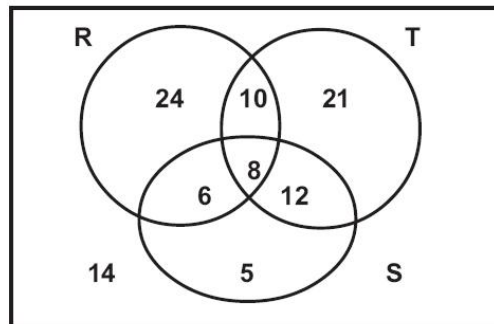
¹Las preguntas 1, 4, 6 y 8 son del autor.

²Pregunta tomada de un documento de análisis de resultado de un simulacro del ECAES realizado en la Escuela de Ingeniería de Antioquia.

³Pregunta reescrita a partir de una pregunta no liberada del ECAES de Ingeniería de Sistemas del año 2008.

- D.) 52
E.) 300

5. Tres de las publicaciones que emite la Universidad se han clasificado de la siguiente manera: publicaciones de investigación: R; publicaciones de deportes: S, y publicaciones de cultura: T. Cien (100) de los estudiantes de la Universidad se han encuestado respecto a la suscripción o suscripciones que poseen de las mencionadas publicaciones; el resultado de la encuesta ha sido representado en el diagrama de Venn⁴.



De acuerdo con el gráfico podría decirse de los estudiantes que:

- I.) 51 han generado preferencia por la publicación cultural
II.) 24 han realizado suscripción a la publicación investigación
III.) 69 no se han interesado por la publicación deportes
IV.) 36 estudiantes están suscritos a las tres publicaciones

Con respecto a las afirmaciones anteriores, ¿cuál de las siguientes es la única respuesta correcta?

- A.) la afirmación I y II son correctas
B.) la afirmación II y III son correctas
C.) la afirmación III y IV son correctas
D.) la afirmación II y IV son correctas
E.) la afirmación I y III son correctas
6. Sean A y B conjuntos no vacíos, ¿cuántos pares ordenados tiene $A \times B$?
- A.) 0
B.) $|A| \cdot |A|$

⁴Enunciado textual de la pregunta 64 del cuadernillo de la segunda sesión del ECAES de Contaduría Pública del año 2004.

- C.) $|A| \cdot |B|$
 D.) $|A| + |B|$
 E.) $MAX\{|A|, |B|\}$

7. Sea el conjunto universal $U = \mathbb{Z}^+$ y los conjuntos:

$$A = \{2, 5, 6, 7, 9\},$$

$$B = \{x \mid x \in \mathbb{Z}^+ \wedge 5 \leq x \leq 15\},$$

$$C = \{x \mid x \in \mathbb{Z}^+ \wedge x < 18 \wedge x \text{ es primo}\}.$$

¿Cuál es el resultado de $C \cap (B \cup \overline{A})$?⁵

- A.) $\{ \}$
 B.) $\{3, 6, 9, 11, 13, 17\}$
 C.) $\{3, 5, 7, 11, 13, 17\}$
 D.) $\{2, 3, 5, 7, 11, 13, 17\}$
 E.) $\{1, 2, 3, 5, 7, 11, 13, 17\}$

8. Sean los conjuntos $A = \{1, 2, 3\}$, $B = \{a, b, c\}$ y $C = \{a, 1, b, 2\}$.

¿Cuál es el resultado de $(C \times C) - (B \times A)$?

- A.) $\{ \}$
 B.) $\{(a, 3), (b, 3), (c, 1), (c, 2), (c, 3)\}$
 C.) $\{(1, a), (1, b), (1, c), (2, a), (2, b), (2, c), (3, a), (3, b), (3, c)\}$
 D.) $\{(a, a), (a, b), (1, a), (1, 1), (1, b), (1, 2), (b, a), (b, b), (2, a), (2, 1), (2, b), (2, 2)\}$
 E.) $\{(a, a), (a, 1), (a, b), (a, 2), (1, a), (1, 1), (1, b), (1, 2), (b, a), (b, 1), (b, b), (b, 2), (2, a), (2, 1), (2, b), (2, 2)\}$

9. ¿Cuál es el conjunto potencia $P(A)$ para el conjunto:

$A = \{x \mid x \text{ es una letra de la palabra "rata"}\}$

- A.) $P(A) = \{ \}$
 B.) $P(A) = \{ \{ \}, \{r\}, \{a\}, \{t\}, \{r, a\}, \{r, t\}, \{a, t\}, \{r, a, t\} \}$
 C.) $P(A) = \{ \{ \}, \{r\}, \{a\}, \{t\}, \{a\}, \{a, r\}, \{r, a\}, \{a, t\}, \{a, a\}, \{t, a\}, \{r, a, t\}, \{r, a, a\}, \{a, t, a\}, \{t, a, r\}, \{r, a, t, a\} \}$
 D.) $P(A) = \{ \{ \}, \{r\}, \{a\}, \{t\}, \{a\} \}$

⁵Las preguntas 7 y 9 han sido reescritas adaptándolas a las condiciones que tienen que cumplir las preguntas tipo ECAES en Colombia, a partir del material de auto evaluación de un libro mexicano de Matemáticas para la Computación de José Alfredo Jiménez [J2009].

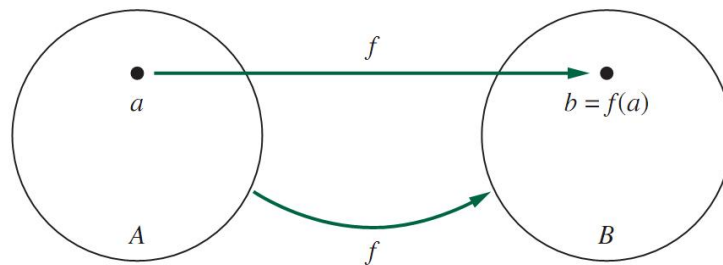
Capítulo 6

Funciones

6.1. Conceptos fundamentales

Definición de función:

Sean A y B conjuntos. Una función f de A a B es una asignación de exactamente un elemento de B a cada elemento de A . Se escribe $f(a) = b$ si b es el único elemento de B asignado por la función f al elemento a de A . Si f es una función de A a B , se escribe $f : A \rightarrow B$.



La función f mapea A en B .

Definición de dominio de una función:

Si f es una función de A a B , se dice que A es el dominio de f y B es el codominio de f . Si $f(a) = b$, se dice que b es la imagen de a y a es la pre-imagen de b . El rango de f es el conjunto de todas las imágenes de A . Así mismo, si f es una función de A a B , se dice que f mapea A en B .

Ejemplo 1:

Sea el conjunto A todas las cadenas de bits de longitud 2 y sea B el conjunto de los números 0, 1 y 2. La función f cuenta la cantidad de ceros que hay en la cadena de bits de longitud 2.

El dominio de la función f es $Dom_f = \{00, 01, 10, 11\}$, $Codom_f = \{0, 1, 2\}$, $f(00) = 2$, 2 es la imagen de 00 y 00 es la pre-imagen de 2.

Ejemplo 2:

Sea $f : \mathbb{Z} \rightarrow \mathbb{N}$ la función que calcula el cuadrado de un número entero. Donde:

$$f(x) = x^2$$

$$Dom_f = \mathbb{Z}$$

$$Codom_f = \mathbb{N}$$

$$Rango_f = \{0, 1, 4, 9, 16, 25, \dots, n^2\}$$

Ejemplo 3:

Cuando se programan funciones en lenguaje C o JAVA se definen el dominio y codominio de la función. Por ejemplo al programar la función piso:

```
int piso(float numero)
{
    -
    -
    -
}
```

El dominio de la función son los reales de precisión sencilla (float) y codominio son los números enteros (int).

Definición de suma y multiplicación de funciones:

Sean f_1 y f_2 funciones de A a \mathbb{R} . Entonces $f_1 + f_2$ y $f_1 \cdot f_2$ son también funciones de A a \mathbb{R} definidas por:

$$(f_1 + f_2)(x) = f_1(x) + f_2(x)$$

$$(f_1 \cdot f_2)(x) = f_1(x) \cdot f_2(x)$$

Ejemplo 4:

f_1 y f_2 son funciones de \mathbb{R} a \mathbb{R} tales que $f_1(x) = x^2$ y $f_2(x) = x - x^2$.

¿Cuáles son las funciones $f_1 + f_2$ y $f_1 \cdot f_2$?

$$(f_1 + f_2)(x) = f_1(x) + f_2(x) = x^2 + x - x^2 = x$$

$$(f_1 \cdot f_2)(x) = f_1(x) \cdot f_2(x) = x^2 \cdot (x - x^2) = x^3 - x^4.$$

Definición:

Sea f una función del conjunto A al conjunto B y sea S un subconjunto de A . La imagen de S es un subconjunto de B que consiste de las imágenes de los elementos de S . Se define la imagen de S por $f(S)$, tal que $f(S) = \{f(s) \mid s \in S\}$.

Ejemplo 5:

Sea $A = \{a, b, c, d, e\}$ y $B = \{1, 2, 3, 4\}$ con $f(a) = 2$, $f(b) = 1$, $f(c) = 4$, $f(d) = 1$ y $f(e) = 1$.

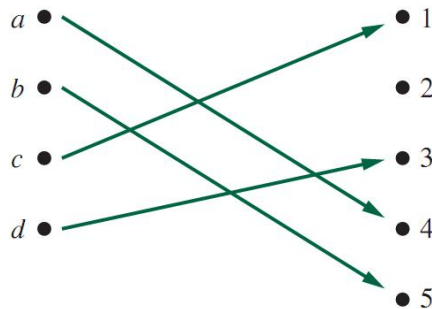
La imagen del subconjunto $S = \{b, c, d\}$ es el conjunto $f(S) = \{1, 4\}$.

6.2. Funciones inyectivas (o funciones uno a uno)

Una función es inyectiva o uno a uno si y únicamente si $f(x) = f(y)$ implica que $x = y$ para toda x y y en el dominio de f .

Ejemplo 6:

Determinar si la función f de $\{a, b, c, d\}$ a $\{1, 2, 3, 4, 5\}$ con $f(a) = 4$, $f(b) = 5$, $f(c) = 1$ y $f(d) = 3$ es inyectiva (uno a uno).



Función inyectiva.

La función f si es inyectiva (uno a uno).

Ejemplo 7:

Determinar si la función $f : \mathbb{Z} \rightarrow \mathbb{Z}$, $f(x) = x^2$ es uno a uno (inyectiva).

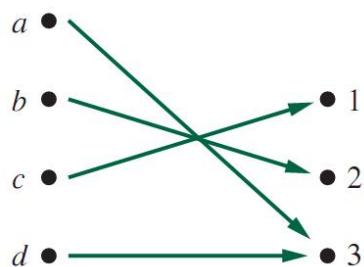
La función f no es uno a uno porque $f(-1) = f(1) = 1$, donde $-1 \neq 1$. Pero, si la función es $f : \mathbb{Z}^+ \rightarrow \mathbb{Z}^+$ entonces sí es uno a uno.

6.3. Funciones sobreyectivas

Una función f de A a B es llamada sobreyectiva si y únicamente si para cada elemento $b \in B$ hay como mínimo un elemento $a \in A$ con $f(a) = b$.

Ejemplo 8:

Sea f la función de $\{a, b, c, d\}$ a $\{1, 2, 3\}$ definida por $f(a) = 3$, $f(b) = 2$, $f(c) = 1$ y $f(d) = 3$. ¿Es f una función sobreyectiva?



Función sobreyectiva.

La función f si es sobreyectiva.

Ejemplo 9:

¿Es la función $f : \mathbb{Z} \rightarrow \mathbb{Z}$, $f(x) = x^2$ sobreyectiva?

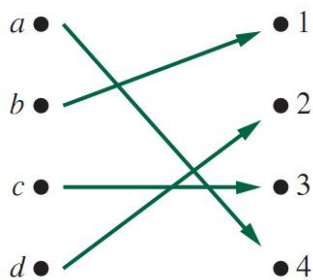
No, porque -1 no es un resultado de f y éste pertenece al conjunto de posibles respuestas ($-1 \in \mathbb{Z}$).

6.4. Funciones biyectivas

La función f tiene la propiedad de biyectividad (o también llamada correspondencia uno a uno) si ésta tiene al mismo tiempo las propiedades de inyectividad y sobreyectividad.

Ejemplo 10:

f es una función biyectiva de $\{a, b, c, d\}$ a $\{1, 2, 3, 4\}$ con $f(a) = 4$, $f(b) = 1$, $f(c) = 3$ y $f(d) = 2$.



Función biyectiva.

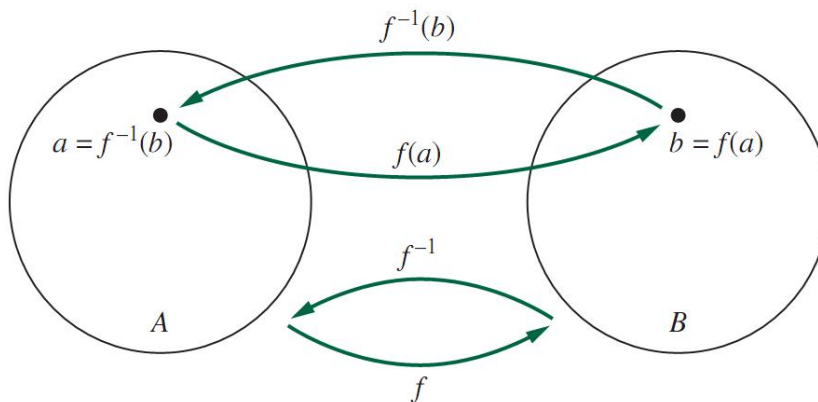
6.5. Funciones inversas y composición de funciones

Definición de función inversa:

Sea f una función biyectiva del conjunto A al conjunto B . La función inversa de f es la función que asigna a un elemento b que pertenece al conjunto B el único elemento a

en el conjunto A tal que $f(a) = b$. La función inversa de f es denotada por f^{-1} , donde, $f^{-1}(b) = a$ si y solo si $f(a) = b$.

Gráficamente:



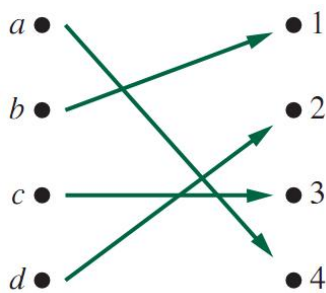
La función f^{-1} es la inversa de la función f .

Si una función f no es biyectiva, entonces no se puede definir una función inversa de f .

Una función biyectiva es también llamada función invertible.

Ejemplo 11:

Sea f la función de $\{a, b, c, d\}$ a $\{1, 2, 3, 4\}$ tal que $f(a) = 4$, $f(b) = 1$, $f(c) = 3$ y $f(d) = 2$. ¿Es f una función invertible, y si es así, cuál es su inversa?



Función invertible.

Al analizar el gráfico se evidencia que f es una función que tiene las propiedades de inyectividad y de sobreyectividad, por lo tanto f es una función con la propiedad de biyectividad. Como f es una función biyectiva entonces se obtiene la siguiente función inversa: $f^{-1}(1) = b$, $f^{-1}(2) = d$, $f^{-1}(3) = c$ y $f^{-1}(4) = a$.

Ejemplo 12:

Sea la función $f : \mathbb{Z} \rightarrow \mathbb{Z}$, donde $f(x) = x + 1$. ¿Es f una función invertible?, si es así, ¿cuál es la función inversa?

f es una función inyectiva y sobreyectiva, por lo tanto es una función biyectiva y es en consecuencia, una función que tiene inversa.

Tenemos que $f(x) = x + 1$, $y = x + 1$, $x = y - 1$. Entonces la función inversa de f es $f^{-1}(y) = y - 1$.

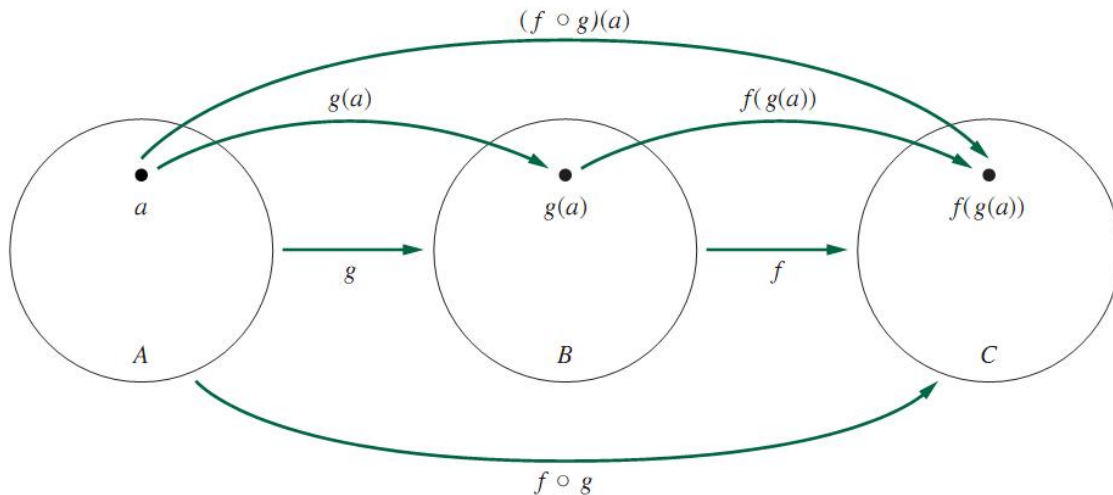
Ejemplo 13:

Sea la función $f : \mathbb{Z} \rightarrow \mathbb{Z}$, donde $f(x) = x^2$. ¿Es f una función invertible?, si es así, ¿cuál es la función inversa?

Se tiene que $f(-1) = (-1)^2 = 1$ y que $f(1) = (1)^2 = 1$, como dos valores diferentes del dominio son mapeados por la función f al mismo valor del codominio, entonces la función f no es una función inyectiva, por lo tanto la función f no es biyectiva y al no ser f una función biyectiva, entonces la función f no tiene inversa.

Definición de composición de funciones:

Sea g una función del conjunto A al conjunto B y sea f una función del conjunto B al conjunto C . La composición de las funciones f y g denotado por $f \circ g$, es la función definida por $(f \circ g)(a) = f(g(a))$, para cada $a \in A$.



Composición de las funciones f y g .

Ejemplo 14:

Sea g la función del conjunto $\{a, b, c\}$ a sí mismo tal que $g(a) = b$, $g(b) = c$ y $g(c) = a$.

Sea f la función del conjunto $\{a, b, c\}$ al conjunto $\{1, 2, 3\}$ tal que $f(a) = 3$, $f(b) = 2$ y $f(c) = 1$. ¿Cuál es la composición de f y g ?, ¿Cuál es la composición de g y f ?

Se da respuesta a la primer pregunta, f compuesto g :

$$(f \circ g)(a) = f(g(a)) = f(b) = 2.$$

$$(f \circ g)(b) = f(g(b)) = f(c) = 1.$$

$$(f \circ g)(c) = f(g(c)) = f(a) = 3.$$

Para la segunda pregunta, g compuesto f no está definido porque el codominio de la función f no es un subconjunto del dominio de la función g , es decir, $\{1, 2, 3\} \not\subseteq \{a, b, c\}$

Ejemplo 15:

Sean f y g las funciones del conjunto de los números enteros al conjunto de los números enteros definidas por $f(x) = 2x + 3$ y $g(x) = 3x + 2$. ¿Cuál es la composición de f y g ?, ¿cuál es la composición de g y f ?

Primero se da respuesta a la pregunta de f compuesto g ,

$$(f \circ g)(x) = f(g(x)) = f(3x + 2) = 2(3x + 2) + 3 = 6x + 4 + 3 = 6x + 7.$$

Ahora se da respuesta a la segunda pregunta con respecto a g compuesto f ,

$$(g \circ f)(x) = g(f(x)) = g(2x + 3) = 3(2x + 3) + 2 = 6x + 9 + 2 = 6x + 11.$$

Con respecto a los resultados anteriores se tiene que $(f \circ g)(x) \neq (g \circ f)(x)$, por lo tanto se evidencia que la propiedad conmutativa no se presenta en la composición de funciones.

Cuando se hace la composición de una función y su inversa, en algún orden, una identidad de la función es obtenida. Primero se debe suponer que f es una función biyectiva del conjunto A al conjunto B . Entonces la función inversa f^{-1} existe y es una función biyectiva del conjunto B al conjunto A . La inversa de la función inversa es la función original, tal que $f^{-1}(b) = a$ donde $f(a) = b$ y $f(a) = b$ donde $f^{-1}(b) = a$. Por lo tanto, $(f^{-1} \circ f)(a) = f^{-1}(f(a)) = f^{-1}(b) = a$, $(f \circ f^{-1})(b) = f(f^{-1}(b)) = f(a) = b$ y $(f^{-1})^{-1} = f$.

6.6. Gráfica de una función

El gráfico de una función ayuda a que ésta sea más fácilmente comprendida, discutiendo si esta es inyectiva, sobreyectiva o biyectiva.

Ejemplo 16:

La siguiente es la gráfica de la función $f(n) = 2n + 1$, donde $f : \mathbb{Z} \rightarrow \mathbb{Z}$, es decir, la función f va del conjunto de los números enteros al conjunto de los números enteros

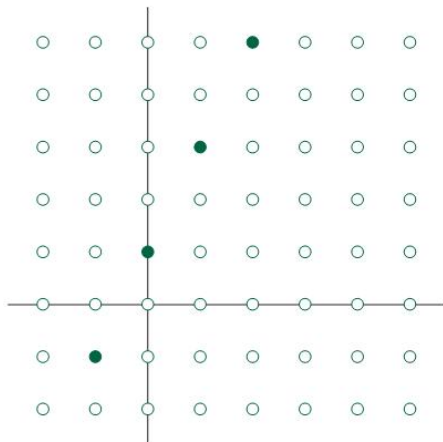


Gráfico de la función $f(n) = 2n + 1$ de $f : \mathbb{Z} \rightarrow \mathbb{Z}$.

En la gráfica fácilmente se puede observar que la función f es inyectiva porque cada número entero en el dominio referencia a un número entero en el codominio que no es referenciado por ningún otro número entero en el dominio. En la gráfica también se puede apreciar que la función f no es sobreyectiva porque hay números enteros del codominio que no son la imagen de algún número entero, tal es el caso de cualquier número entero par el cual no es referenciado por ningún número entero. Por lo tanto la función f no es biyectiva.

6.7. Ejercicios

1. Sean los conjuntos $A = \{2, 3, 4, 5\}$ y $B = \{6, 7, 8, 9, 10\}$, con $|A| = 4$ y $|B| = 5$, determinar
 - a.) ¿Cuántas funciones hay de A a B ?
 - b.) ¿Cuántas funciones $f : A \rightarrow B$ cumplen $f(2) = 6$?
 - c.) ¿Cuántas funciones hay de B a A ?
 - d.) ¿Cuántas funciones $g : B \rightarrow A$ cumplen $g(10) = 5$ y $g(8) = 3$?
 - e.) ¿se obtiene el mismo valor en las dos preguntas anteriores?
 - f.) ¿Cuántas funciones inyectivas hay de A a B ?
 - g.) ¿Cuántas funciones sobreyectivas hay de A a B ?
 - h.) ¿Cuántas funciones biyectivas hay de A a B ?
 - i.) ¿Cuántas funciones inyectivas hay de B a A ?

- j.) ¿Cuántas funciones sobreyectivas hay de B a A^1 ?
- k.) ¿Cuántas funciones biyectivas hay de B a A ?
- Si hay 2187 funciones $f : A \rightarrow B$ y $|B| = 3$, ¿cuál es el valor de $|A|$?
 - Si $A = \{1, 2, 3, 4, 5\}$ y hay 6720 funciones inyectivas $f : A \rightarrow B$, ¿cuál es el valor de $|B|$?
 - Encontrar el rango y el dominio de la función que asigna a cada entero positivo su último dígito.
 - Encontrar el rango y el dominio de la función que asigna a cada entero positivo la multiplicación de sus dígitos.
 - Encontrar el rango y el dominio de la función f que asigna a cada entero positivo la suma de sus dígitos. Por ejemplo $f(1) = 1$, $f(11) = 1 + 1 = 2$ y $f(532) = 5 + 3 + 2 = 10$.
 - Encontrar el rango y el dominio de la función que asigna a cada entero positivo el entero que le sigue (es decir, su sucesor.)
 - Encontrar el rango y el dominio de la función que asigna a una cadena de bits la cantidad de unos que esta contiene.
 - Sean las siguientes funciones $F_i : \mathbb{Z} \times \mathbb{Z} \rightarrow \mathbb{Z}$

$$F_1(m, n) = |4m| - |2n|$$

$$F_2(m, n) = m^3 - n^3$$

$$F_3(m, n) = 3m - n$$

$$F_4(m, n) = m^2 - n$$

$$F_5(m, n) = 2m - n$$

$$F_6(m, n) = m^2 - n^2$$

$$F_7(m, n) = |m| - |n|$$

¿Cuáles de las funciones anteriores son uno a uno?, ¿cuáles son sobreyectivas? y ¿cuáles son biyectivas?

- Sea la función $F(m, n) = |m| \cdot |n|$ donde $F : \mathbb{Z} \times \mathbb{Z} \rightarrow \mathbb{Z}$, ¿La función es uno a uno?, ¿la función es sobreyectiva?, y ¿la función es biyectiva?
- Sea la función $F(m, n) = 3m - |n|$ donde $F : \mathbb{Z} \times \mathbb{Z} \rightarrow \mathbb{Z}$, ¿La función es uno a uno?, ¿la función es sobreyectiva?, y ¿la función es biyectiva?
- Sea la función $F(m, n) = n$ donde $F : \mathbb{Z} \times \mathbb{Z} \rightarrow \mathbb{Z}$, ¿La función es uno a uno?, ¿la función es sobreyectiva?, y ¿la función es biyectiva?

¹**Ayuda:** Leer la sección: “*Funciones Suprayectivas: Números de Stirling de Segundo Tipo*” del libro: “*Matemáticas Discreta y Combinatoria*” de Ralph P. Grimaldi, en donde se presenta una buena explicación del conteo de funciones sobreyectivas (o suprainyectivas.)

13. Sea la función $F(m, n) = m \cdot n$ donde $F : \mathbb{Z} \times \mathbb{Z} \rightarrow \mathbb{Z}$, ¿La función es uno a uno?, ¿la función es sobreyectiva?, y ¿la función es biyectiva?
14. ¿Cuántas funciones biyectivas hay del conjunto A al conjunto B si $|A| = |B|$?
15. Sea la función $f : \mathbb{N} \rightarrow \mathbb{N} \times \mathbb{N}$,

$$f(0) = (x_0, y_0) = (0, 0)$$

$$f(n+1) = (x_{n+1}, y_{n+1}) = \begin{cases} (x_n - 1, y_n + 1) & \text{si } x_n \geq 1 \\ (y_n + 1, 0) & \text{si } x_n = 0 \end{cases}$$

Determinar:

- a.) ¿Qué es $f(8)$?
 - b.) ¿Es f una función inyectiva?
 - c.) ¿Es f una función sobreyectiva?
 - d.) ¿Es f una función biyectiva?, si esto es así, ¿entonces el conjunto \mathbb{N} tiene la misma cardinalidad que el conjunto $\mathbb{N} \times \mathbb{N}$?
 - e.) ¿Es f una función invertible?, si es así, ¿cuál es la inversa de f ?
16. Sea $C = \{0, 1, 00, 01, 10, 11, 000, 001, 010, 011, 100, 101, 110, 111, \dots\}$, donde C es el conjunto de todas las cadenas que se pueden formar con ceros y unos. Sea la función $f : C \times C \rightarrow C$ donde $f(s, r) = s00r$, lo que hace la función f es concatenar dos cadenas s y r colocando en medio de ellas la cadena 00.

Determinar:

- a.) ¿Qué es $f(001, 110)$?
- b.) ¿Es f una función inyectiva?
- c.) ¿Es f una función sobreyectiva?
- d.) ¿Es f una función biyectiva?
- e.) ¿Es f una función invertible?, si es así, ¿cuál es la inversa de f ?

6.8. Preguntas tipo ECAES

1. Sea la siguiente definición recursiva, donde n es un número entero no negativo² :

$$f(n) = \begin{cases} n & \text{si } n = 0 \text{ o } n = 1 \\ f(n-2) + f(n-1) & \text{si } n > 1 \end{cases}$$

¿Cuál es el resultado de $f(5)$?

- A.) 2
- B.) 3
- C.) 4
- D.) 5
- E.) 6

2. Sea la siguiente definición recursiva, donde n es un número entero no negativo:

$$f(n) = \begin{cases} n & \text{si } n = 0 \text{ o } n = 1 \\ g(n, 2, 0, 1) & \text{si } n > 1 \end{cases}$$

$$g(n, i, a, b) = \begin{cases} a + b & \text{si } i = n \\ g(n, i+1, b, a+b) & \text{si } i < n \end{cases}$$

¿Cuál es el resultado de $f(5)$?

- A.) 2
- B.) 3
- C.) 4
- D.) 5
- E.) 6

²Las preguntas 1, 2, 3, 4, 7, 8, 9, 10, 11, 12, 14, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25 y 26 son del autor.

3. Sea la siguiente función recursiva, donde n es un número entero no negativo:

$$f(n) = \begin{cases} 3 & \text{si } n = 0 \\ -2 & \text{si } n = 1 \\ 3 \cdot f(n-2) - 2 \cdot f(n-1) & \text{si } n > 1 \end{cases}$$

¿Cuál es el valor de $f(4)$?

- A.) 10
- B.) 18
- C.) -32
- D.) 103
- E.) 200

4. Sea la siguiente definición recursiva, donde n es un número entero no negativo:

$$f(n) = \begin{cases} 3 & \text{si } n = 0 \\ -2 & \text{si } n = 1 \\ g(n, 2, 3, -2) & \text{si } n > 1 \end{cases}$$

$$g(n, i, a, b) = \begin{cases} 3 \cdot a - 2 \cdot b & \text{si } i = n \\ g(n, i+1, b, 3 \cdot a - 2 \cdot b) & \text{si } i < n \end{cases}$$

¿Cuál es el resultado de $f(4)$?

- A.) 10
- B.) 18
- C.) -32
- D.) 103
- E.) 200

5. Sea la siguiente definición recursiva³ :

³Las preguntas 5 y 6 han sido reescritas a partir del ejemplo 6 de la Guía de Orientación del ECAES de Ingeniería de Sistemas del año 2010 [MI2010].

$$f(m, n) = \begin{cases} 1 & \text{si } n = 0 \\ m \cdot f(m, n - 1) & \text{si } n > 0 \end{cases}$$

¿Cuál es el resultado de $f(2, 3)$?

- A.) 8
- B.) 9
- C.) 6
- D.) 12
- E.) 2

6. Sea la siguiente definición recursiva:

$$f(m, n) = \begin{cases} 1 & \text{si } n = 0 \\ m \cdot f(m, n - 1) & \text{si } n > 0 \end{cases}$$

¿Cuál es el resultado correcto para $f(m, n)$?

- A.) n^m
- B.) m^n
- C.) $m \cdot n$
- D.) $n!$
- E.) $n \cdot m!$

7. Sea la siguiente definición recursiva:

$$f(m, n) = \begin{cases} 1 & \text{si } n = 0 \\ n \cdot f(m, n - 1) & \text{si } n > 0 \end{cases}$$

¿Cuál es el resultado de $f(2, 3)$?

- A.) 8
- B.) 9
- C.) 6

D.) 12

E.) 2

8. Sea la siguiente definición recursiva:

$$f(m, n) = \begin{cases} 1 & \text{si } n = 0 \\ n \cdot f(m, n - 1) & \text{si } n > 0 \end{cases}$$

¿Cuál es el resultado correcto para $f(m, n)$?

A.) n^m B.) m^n C.) $m \cdot n$ D.) $n!$ E.) $m!$

9. Para m y n enteros no negativos, la función M se define como:

$$M(m, n) = \begin{cases} n & \text{si } m = 0 \\ M(m - 1, n + 1) & \text{si } m > 0 \end{cases}$$

¿Cuál es el resultado para $M(3, 4)$?

A.) 3

B.) 4

C.) 5

D.) 6

E.) 7

10. Para m y n enteros no negativos, la función $a(m, n)$ se define como:

$$a(m, n) = \begin{cases} 0 & \text{si } m = 0 \text{ y } n = 0 \\ a(m - 1, n) + 1 & \text{si } m > 0 \text{ y } n = 0 \\ a(m, n - 1) + 1 & \text{si } n > 0 \end{cases}$$

¿Cuál es el resultado para $a(2, 3)$?

- A.) 6
- B.) 5
- C.) 4
- D.) 3
- E.) 2

11. Para m y n enteros no negativos, la función $P(m, n)$ se define como:

$$P(m, n) = \begin{cases} 1 & \text{si } m = 1 \\ 1 & \text{si } n = 1 \\ P(m, m) & \text{si } m < n \\ P(m, m-1) + 1 & \text{si } m = n \text{ y } n > 1 \\ P(m, n-1) + P(m-n, n) & \text{si } m > n > 1 \end{cases}$$

¿Cuál es el resultado para $P(5, 5)$?

- A.) 7
- B.) 6
- C.) 8
- D.) 4
- E.) 9

12. Para m y n enteros no negativos, la función $a(m, n)$ se define como:

$$a(m, n) = \begin{cases} 0 & \text{si } m = 0 \text{ y } n = 0 \\ a(m-1, n) + 1 & \text{si } m > 0 \text{ y } n = 0 \\ a(m, n-1) + n & \text{si } n > 0 \end{cases}$$

¿Cuál es el resultado para $a(2, 3)$?

- A.) 5
- B.) 6
- C.) 7
- D.) 8
- E.) 9

13. Para m y n enteros no negativos, la función de Ackerman (A) se define como⁴ :

$$A(m, n) = \begin{cases} n + 1 & \text{si } m = 0 \\ A(m - 1, 1) & \text{si } n = 0 \\ A(m - 1, A(m, n - 1)) & \text{si } m > 0 \text{ y } n > 0 \end{cases}$$

¿Cuál es el resultado para $A(1, 4)$?

- A.) 3
- B.) 4
- C.) 5
- D.) 6
- E.) 7

14. Para n y m enteros no negativos, la función de Morris (M) se define como:

$$M(m, n) = \begin{cases} n + 1 & \text{si } m = n \\ M(m, M(m - 1, n + 1)) & \text{si } m > n \text{ y } n \geq 0 \end{cases}$$

¿Cuál es el resultado para $M(4, 0)$?

- A.) 3
- B.) 4
- C.) 5
- D.) 6
- E.) 7

15. Para n que es un entero no negativo, la función f se define como⁵ :

$$f(0) = (x_0, y_0) = (0, 0)$$

$$f(n + 1) = (x_{n+1}, y_{n+1}) = \begin{cases} (x_n - 1, y_n + 1) & \text{si } x_n \geq 1 \\ (y_n + 1, 0) & \text{si } x_n = 0 \end{cases}$$

⁴Pregunta reescrita a partir de una pregunta no liberada del ECAES de Ingeniería de Sistemas del año 2005.

⁵Pregunta reescrita a partir de una pregunta no liberada del ECAES de Ingeniería de Sistemas del año 2006.

¿Cuál es el resultado para $f(4)$?

- A.) (1, 0)
- B.) (0, 1)
- C.) (2, 0)
- D.) (1, 1)
- E.) (0, 2)

16. Sea la siguiente definición recursiva, donde $x \bmod y$ es el residuo de la división entera de x por y :

$$M(x, y) = \begin{cases} y & \text{si } (y \leq x) \text{ y } (x \bmod y) = 0 \\ M(y, x) & \text{si } x < y \\ M(y, (x \bmod y)) & \text{de otro modo} \end{cases}$$

¿Cuál es el resultado de $M(91, 221)$?

- A.) 2
- B.) 3
- C.) 13
- D.) 18
- E.) 27

17. Para m y n enteros no negativos, donde $m \geq n$, la función de Stirling (S) se define como:

$$S(m, n) = \begin{cases} 1 & \text{si } n = 1 \\ 1 & \text{si } m = n \\ S(m-1, n-1) + n \cdot S(m-1, n) & \text{si } m > n \end{cases}$$

¿Cuál es el resultado de $S(4, 2)$?

- A.) 13
- B.) 7
- C.) 12
- D.) 4
- E.) 15

18. Para n y r enteros no negativos, donde $n \geq r$, la función Combinatorio (C) se define como:

$$C(n, r) = \begin{cases} 1 & \text{si } r = 0 \\ 1 & \text{si } n = r \\ n & \text{si } r = 1 \\ C(n-1, r) + C(n-1, r-1) & \text{si } n > r > 1 \end{cases}$$

¿Cuál es el resultado de $C(4, 2)$?

- A.) 6
 - B.) 4
 - C.) 10
 - D.) 5
 - E.) 15
19. Sean las siguientes funciones $F_i : \mathbb{Z} \times \mathbb{Z} \rightarrow \mathbb{Z}$

- $F_1(m, n) = |4m| + |2n|$
- $F_2(m, n) = m^3 - n^3$
- $F_3(m, n) = 3m - n$
- $F_4(m, n) = m^2 - n$

¿Cuál de las siguientes opciones es la única respuesta correcta con respecto a la propiedad de sobreyectiva en las funciones anteriores?

- A.) Únicamente F_1 y F_2
 - B.) Únicamente F_2 y F_3
 - C.) Únicamente F_3 y F_4
 - D.) Únicamente F_2 y F_4
 - E.) Únicamente F_1 y F_3
20. Sean las siguientes funciones $F_i : \mathbb{Z} \times \mathbb{Z} \rightarrow \mathbb{Z}$

- $F_1(m, n) = 3m - n$
- $F_2(m, n) = m^3 - n^3$
- $F_3(m, n) = |4m| - |2n|$
- $F_4(m, n) = m^2 - n$

¿Cuál de las siguientes opciones es la única respuesta correcta con respecto a la propiedad de biyectividad en las funciones anteriores?

- A.) Únicamente F_1 y F_2
- B.) Únicamente F_2 y F_3
- C.) Únicamente F_3 y F_4
- D.) Únicamente F_1 y F_3
- E.) Ninguna función es biyectiva

Sean los conjuntos $A=\{1, 3, 5, 7, 9\}$ y $B=\{0, 2, 4, 6, 8, 10\}$, con los cuales se deben trabajar desde el ejercicio 21 al 25.

21. Cuántas funciones hay de A a B ?

- A.) 0
- B.) $|A|!$
- C.) $|A|^{|A|}$
- D.) $|B|^{|A|}$
- E.) $|A|^{|B|}$

22. Cuántas funciones inyectivas hay de A a B ?

- A.) 0
- B.) $|A|!$
- C.) $|A|! - 1$
- D.) $|B|!$
- E.) $|B|! - 1$

23. Cuántas funciones inyectivas hay de B a A ?

- A.) 0
- B.) $|A|!$
- C.) $|A|! - 1$
- D.) $|B|!$
- E.) $|B|! - 1$

24. Cuántas funciones sobreyectivas hay de A a B ?

- A.) 0
- B.) $|A|!$
- C.) $|A|^{|A|}$
- D.) $|B|^{|A|}$
- E.) $|A|^{|B|}$

25. Cuántas funciones biyectivas hay de A a B ?

- A.) 0
- B.) $|A|!$
- C.) $|A|! - 1$
- D.) $|B|!$
- E.) $|B|! - 1$

26. Sean A y B conjuntos no vacíos, ¿cuántas funciones biyectivas hay del conjunto A al conjunto B si $|A| \neq |B|$?

- A.) 0
- B.) $|A|$
- C.) $|A| \cdot |B|$
- D.) $|A|!$
- E.) $|A| \cdot |A|!$

27. Sea $C = \{0, 1, 00, 01, 10, 11, 000, 001, 010, 011, 100, 101, 110, 111, \dots\}$, C es el conjunto de todas las cadenas que se pueden formar con ceros y unos. Sea la función $f : C \times C \rightarrow C$ donde $f(s, r) = s00r$, lo que hace la función f es concatenar dos cadenas s y r colocando en medio de ellas la cadena 00.

¿Qué es lo único que se puede afirmar de la función f ?⁶

- A.) La función f es biyectiva porque permite concatenar cualquier par de cadenas de ceros y unos.
- B.) La función f no es inyectiva porque para diferentes pares de cadenas la función f da la misma cadena de resultado.
- C.) La función f es sobreyectiva porque cualquier cadena de ceros y unos es obtenida como resultado de la función.
- D.) La función f no es biyectiva porque no cumple la inyectividad pero si es sobreyectiva.

⁶Pregunta reescrita a partir de una pregunta no liberada del ECAES de Ingeniería de Sistemas del año 2008.

Capítulo 7

Relaciones

7.1. Relación binaria

Sean A y B conjuntos. Una relación binaria de A a B es un subconjunto de $A \times B$. En otras palabras, una relación binaria de A a B es un conjunto R de pares ordenados donde el primer elemento de cada par ordenado pertenece a A y el segundo pertenece a B . Se utiliza la notación aRb para denotar que $(a, b) \in R$ y $a \not R b$ para denotar que $(a, b) \notin R$.

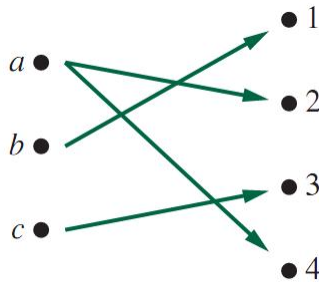
Ejemplo 1:

Sea A el conjunto de estudiantes de Ingeniería de Sistemas de la Institución, y sea B el conjunto de materias de Ingeniería de Sistemas de la institución. Sea R la relación que consiste de todos los pares ordenados (a, b) donde a es un estudiante que tiene matriculada la materia b . Si los estudiantes “Jaime” y “Claudia” están matriculados en la materia “Matemáticas Discretas” entonces los pares ordenados $(Jaime, Matemáticas Discretas)$ y $(Claudia, Matemáticas Discretas)$ pertenecen a la relación R , adicionalmente “Claudia” también tiene matriculado el curso de “Bases de Datos”, lo cual se representa con el par ordenado $(Claudia, Bases de Datos)$, que también pertenece a la relación R .

Ejemplo 2:

Sea $A = \{a, b, c\}$ y $B = \{1, 2, 3, 4\}$. Entonces $\{(a, 2), (a, 4), (b, 1), (c, 3)\}$ es una relación de A a B , la cual también se puede representar como $aR2$, $aR4$, $bR1$ y $cR3$. Esta relación puede ser fácilmente representada de forma gráfica y tabular de la siguiente forma:

Gráficamente:



Tabularmente:

R	1	2	3	4
a		X		X
b	X			
c			X	

7.2. Funciones como relaciones

Si R es una relación del conjunto A al conjunto B tal que cada elemento de A es la primera componente de exactamente un par ordenado de R , entonces una función puede ser definida con la relación R . De esta forma se evidencia que las relaciones son una generalización de las funciones y sirven para representar una clase más amplia de relaciones entre conjuntos.

7.3. Relaciones en un conjunto

Las relaciones de un conjunto A consigo mismo son de gran interés.

Definición de relación:

Una relación en el conjunto A es una relación del conjunto A al conjunto A , es decir, una relación en un conjunto A es un subconjunto de $A \times A$.

Ejemplo 3:

Sea $A = \{1, 2, 3, 4\}$ y la relación $R = \{(a, b) \mid a \text{ divide } b\}$ definida sobre el conjunto A . ¿Cuáles pares ordenados pertenecen a la relación R ?

El producto cruz del conjunto A es el siguiente:

$$A \times A = \{(1, 1), (1, 2), (1, 3), (1, 4), \\ (2, 1), (2, 2), (2, 3), (2, 4), \\ (3, 1), (3, 2), (3, 3), (3, 4), \\ (4, 1), (4, 2), (4, 3), (4, 4)\}$$

y los pares ordenados de este producto cruz donde la primera componente divide a la segunda son: $(1, 1)$, $(1, 2)$, $(1, 3)$, $(1, 4)$, $(2, 2)$, $(2, 4)$, $(3, 3)$, y $(4, 4)$, los cuales conforman la relación R

$$R = \{(1, 1), (1, 2), (1, 3), (1, 4), (2, 2), (2, 4), (3, 3), (4, 4)\}.$$

Observe que $R \subseteq A \times A$.

Ejemplo 4:

Considerar las siguientes relaciones definidas en el conjunto de los números enteros:

$$\begin{aligned} R_1 &= \{(a, b) \mid a \leq b\} \\ R_2 &= \{(a, b) \mid a > b\} \\ R_3 &= \{(a, b) \mid a = b \vee a = -b\} \\ R_4 &= \{(a, b) \mid a = b\} \\ R_5 &= \{(a, b) \mid a = b + 1\} \\ R_6 &= \{(a, b) \mid a + b \leq 3\} \end{aligned}$$

¿Cada uno de los siguientes pares ordenados $(1, 1)$, $(1, 2)$, $(2, 1)$, $(1, -1)$ y $(2, 2)$ pertenecen a cuáles relaciones?

El par ordenado $(1, 1)$ se encuentra presente en las relaciones R_1 , R_3 , R_4 y R_6 , el par ordenado $(1, 2)$ se encuentra presente en las relaciones R_1 y R_6 , el par ordenado $(2, 1)$ se encuentra presente en las relaciones R_2 , R_5 y R_6 , el par ordenado $(1, -1)$ se encuentra presente en las relaciones R_2 , R_3 y R_6 y el par ordenado $(2, 2)$ se encuentra presente en las relaciones R_1 , R_3 y R_4 .

Ejemplo 5:

¿Cuántas relaciones hay en un conjunto de n elementos?

Una relación en un conjunto A es un subconjunto de $A \times A$, donde $A \times A$ tiene n^2 pares ordenados cuando $|A| = n$. Recordar que un conjunto con m elementos tiene 2^m subconjuntos, entonces $2^{|A \times A|} = 2^{n^2}$ es el número total de relaciones del conjunto A al conjunto A .

Por ejemplo si $A = \{a, b, c\}$ entonces hay $2^{|A|^2} = 2^{3^2} = 2^9 = 512$ relaciones en el conjunto A .

Ejemplo 6:

¿Cuántas y cuáles relaciones hay en el conjunto $A = \{1, 2\}$?

Primero se debe calcular el producto cruz del conjunto A con el mismo:

$$A \times A = \{(1, 1), (1, 2), (2, 1), (2, 2)\}$$

Luego, como una relación es cualquier subconjunto que se puede sacar del producto cruz del conjunto que se está trabajando, entonces se debe comenzar con los subconjuntos de tamaño cero, tamaño uno, tamaño dos y así sucesivamente hasta el subconjunto que tiene la misma cardinalidad de $A \times A$.

$$\begin{aligned} R_1 &= \{\} \\ R_2 &= \{(1, 1)\} \\ R_3 &= \{(1, 2)\} \\ R_4 &= \{(2, 1)\} \\ R_5 &= \{(2, 2)\} \\ R_6 &= \{(1, 1), (1, 2)\} \\ R_7 &= \{(1, 1), (2, 1)\} \\ R_8 &= \{(1, 1), (2, 2)\} \\ R_9 &= \{(1, 2), (2, 1)\} \\ R_{10} &= \{(1, 2), (2, 2)\} \\ R_{11} &= \{(2, 1), (2, 2)\} \\ R_{12} &= \{(1, 1), (1, 2), (2, 1)\} \\ R_{13} &= \{(1, 1), (1, 2), (2, 2)\} \\ R_{14} &= \{(1, 1), (2, 1), (2, 2)\} \\ R_{15} &= \{(1, 2), (2, 1), (2, 2)\} \\ R_{16} &= \{(1, 1), (1, 2), (2, 1), (2, 2)\} \end{aligned}$$

Son 16 relaciones las cuales coinciden con el resultado que se obtiene al utilizar la fórmula que se trabajó en el ejemplo anterior, $2^{|A|^2} = 2^{2^2} = 2^4 = 16$.

7.4. Propiedades de las relaciones**7.4.1. Propiedad de reflexividad**

Una relación R en un conjunto A tiene la propiedad de reflexividad si $(a, a) \in R$ para cada elemento $a \in A$.

Ejemplo 7:

Considerar las siguientes relaciones en el conjunto $A = \{1, 2, 3, 4\}$

$$R_1 = \{(1, 1), (1, 2), (2, 1), (2, 2), (3, 4), (4, 1), (4, 4)\}$$

$$R_2 = \{(1, 1), (1, 2), (2, 1)\}$$

$$R_3 = \{(1, 1), (1, 2), (1, 4), (2, 1), (2, 2), (2, 4), (3, 3), (4, 1), (4, 2), (4, 4)\}$$

$$R_4 = \{(2, 1), (3, 1), (3, 2), (4, 1), (4, 2), (4, 3)\}$$

$$R_5 = \{(1, 1), (1, 2), (1, 3), (1, 4), (2, 2), (2, 3), (2, 4), (3, 3), (3, 4), (4, 4)\}$$

$$R_6 = \{(3, 4)\}$$

¿Cuáles de estas relaciones tienen la propiedad de reflexividad?

Se tiene que entrar a analizar que se encuentren presentes todos los siguientes pares ordenados: $(1, 1)$, $(2, 2)$, $(3, 3)$, $(4, 4)$. Estos pares ordenados se encuentran todos presentes únicamente en las relaciones R_3 y R_5 , las cuales son relaciones que tienen la propiedad de reflexividad.

7.4.2. Propiedad de simetría

Una relación R en un conjunto A tiene la propiedad de simetría si cuando $(a, b) \in R$ entonces también $(b, a) \in R$, para $a, b \in A$.

Ejemplo 8:

¿Cuáles de las relaciones del ejemplo 7 anterior tienen la propiedad de simetría?

Las únicas relaciones que tienen la propiedad de simetría son R_2 y R_3 .

7.4.3. Propiedad de antisimetría

Una relación R en un conjunto A tiene la propiedad de antisimetría si $(a, b) \in R$ y $a \neq b$ entonces $(b, a) \notin R$, para $a, b \in A$.

Ejemplo 9:

¿Cuáles de las relaciones del ejemplo 7 tienen la propiedad de antisimetría?

Las únicas relaciones que tienen la propiedad de antisimetría son R_4 , R_5 y R_6 .

7.4.4. Propiedad de transitividad

Una relación R en un conjunto A tiene la propiedad de transitividad si cuando se presentan $(a, b) \in R$ y $(b, c) \in R$ también se presenta $(a, c) \in R$, para $a, b, c \in A$.

Ejemplo 10:

¿Cuáles de las relaciones del ejemplo 7 tienen la propiedad de transitividad?

Las únicas relaciones que tienen la propiedad de transitividad son R_3 , R_4 , R_5 y R_6 .

Ejemplo 11:

Considerar la siguiente relación definida en el conjunto de los números enteros:

$$R = \{(a, b) \mid a \geq b\}$$

¿La relación R cumple la propiedad de reflexividad?, ¿cumple la propiedad de simetría?, ¿cumple la propiedad de antisimetría?, y, ¿cumple la propiedad de transitividad?

R cumple la propiedad de reflexividad porque cualquier número entero es mayor o igual a el mismo. R no cumple la propiedad de simetría porque hay pares ordenados que pertenecen a la relación para los cuales el par ordenado simétrico no pertenece a la relación, tal es el caso del par ordenado $(2, 1) \in R$ porque $2 \geq 1$ pero $(1, 2) \notin R$ porque $1 \not\geq 2$. La relación R cumple la propiedad de antisimetría porque la única posibilidad de que $a \geq b$ y $b \geq a$ es que $a = b$, de esta forma $(a, b) \in R$ y $(b, a) \in R$ si y solo si $a = b$. La relación R también cumple la propiedad de transitividad ya que para cualquier conjunto de tres números enteros a, b y c , si $a \geq b$ y $b \geq c$ entonces se cumple que $a \geq c$, con lo cual se tiene que si $(a, b) \in R$ y $(b, c) \in R$ entonces $(a, c) \in R$.

Ejemplo 12:

Considerar la siguiente relación definida en el conjunto de los números enteros:

$$R = \{(a, b) \mid a = b\}$$

¿La relación R cumple la propiedad de reflexividad?, ¿cumple la propiedad de simetría?, ¿cumple la propiedad de antisimetría?, ¿cumple la propiedad de transitividad?, y, ¿es posible que al mismo tiempo una relación tenga las propiedades de simetría y antisimetría?

La relación R cumple la propiedad de reflexividad porque cualquier número entero es igual a el mismo. R cumple la propiedad de simetría porque todos los pares ordenados que pertenecen a la relación R son de la forma (a, a) para $a \in \mathbb{Z}$ con lo cual todo par ordenado es simétrico con el mismo. R cumple la propiedad de antisimetría porque todos los pares ordenados que pertenecen a la relación R son de la forma (a, a) para $a \in \mathbb{Z}$ con lo cual se garantiza la definición de antisimetría donde $a = a$. La relación R también cumple la propiedad de transitividad ya que para cualquier conjunto de tres números enteros a, b y c , si $(a, b) \in R$ y $(b, c) \in R$ es porque $a = b$ y $b = c$, de donde se obtiene que $a = c$ y que por lo tanto $(a, c) \in R$, con lo cual se cumple la definición de la propiedad de transitividad.

Con respecto a la última pregunta, ¿es posible que al mismo tiempo una relación tenga las propiedades de simetría y antisimetría?, la respuesta es que sí y este ejemplo es uno de esos casos, es de vital importancia recalcar que dichas propiedades no son excluyentes en una relación, donde las dos pueden estar presentes o las dos pueden estar ausentes

al mismo tiempo.

7.5. Combinación de relaciones

Como las relaciones de A a B son subconjuntos del conjunto de pares ordenados $A \times B$, entonces dos relaciones de A a B pueden ser combinadas (u operadas) de la forma en que se combinan dos conjuntos.

Ejemplo 13:

Sean los conjuntos $A = \{1, 2, 3\}$ y $B = \{1, 2, 3, 4\}$. Las siguientes relaciones R_1 y R_2 están definidas del conjunto A al conjunto B , donde $R_1 = \{(1, 1), (2, 2), (3, 3)\}$ y $R_2 = \{(1, 1), (1, 2), (1, 3), (1, 4)\}$, estas relaciones pueden ser combinadas obteniendo:

$$\begin{aligned} R_1 \cup R_2 &= \{(1, 1), (1, 2), (1, 3), (1, 4), (2, 2), (3, 3)\} \\ R_1 \cap R_2 &= \{(1, 1)\} \\ R_1 - R_2 &= \{(2, 2), (3, 3)\} \\ R_2 - R_1 &= \{(1, 2), (1, 3), (1, 4)\} \end{aligned}$$

Ejemplo 14:

Sean las siguientes relaciones R_1 y R_2 que están definidas en el conjunto de los números enteros, donde $R_1 = \{(x, y) \mid x < y\}$ y $R_2 = \{(x, y) \mid x > y\}$.

¿Qué es $R_1 \cup R_2$, $R_1 \cap R_2$, $R_1 - R_2$ y $R_2 - R_1$?

$$\begin{aligned} R_1 \cup R_2 &= \{(x, y) \mid x \neq y\} \\ R_1 \cap R_2 &= \{\} \\ R_1 - R_2 &= R_1 \\ R_2 - R_1 &= R_2 \end{aligned}$$

7.6. Composición y potencia de relaciones

Definición de composición de relaciones:

Sea R una relación del conjunto A al conjunto B y sea S una relación del conjunto B al conjunto C . La composición de las relaciones R y S es la relación consistente de los pares ordenados (a, c) , donde $a \in A$ y $c \in C$ y para los cuales existe un elemento $b \in B$ tales que $(a, b) \in R$ y $(b, c) \in S$. Se denota la composición de R y S por $S \circ R$.

Ejemplo 15:

Sean los conjuntos $A = \{1, 2, 3\}$, $B = \{1, 2, 3, 4\}$, $C = \{0, 1, 2\}$ y las relaciones $R = \{(1, 1), (1, 4), (2, 3), (3, 1), (3, 4)\}$ del conjunto A al conjunto B y $S = \{(1, 0), (2, 0), (3, 1), (3, 2), (4, 1)\}$ del conjunto B al conjunto C . ¿Qué es $S \circ R$?

$$S \circ R = \{(1, 0), (1, 1), (2, 1), (2, 2), (3, 0), (3, 1)\}$$

Ejemplo 16:

Sea R la relación en el conjuntos de las personas del mundo tal que $(a, b) \in R$ si a es el padre de b . ¿Qué da como resultado $R \circ R$?

$R \circ R$ da como resultado una relación que es el conjunto de pares ordenados de personas donde la primer componente es el abuelo paterno de la segunda componente.

Definición de potencia de una relación:

Sea R una relación en el conjunto A . La *potencia* R^n , donde $n \in \mathbb{Z}^+$, es definida recursivamente por

$$R^n = \begin{cases} R & \text{si } n = 1 \\ R^{n-1} \circ R & \text{si } n > 1 \end{cases}$$

Ejemplo 17:

Sea la relación $R = \{(1, 1), (2, 1), (3, 2), (4, 3)\}$. ¿Cuál es la potencia R^n para $n \geq 1$?

$$\begin{aligned} R^1 &= R = \{(1, 1), (2, 1), (3, 2), (4, 3)\} \\ R^2 &= R^1 \circ R = \{(1, 1), (2, 1), (3, 2), (4, 3)\} \circ \{(1, 1), (2, 1), (3, 2), (4, 3)\} \\ &= \{(1, 1), (2, 1), (3, 1), (4, 2)\} \\ R^3 &= R^2 \circ R = \{(1, 1), (2, 1), (3, 1), (4, 2)\} \circ \{(1, 1), (2, 1), (3, 2), (4, 3)\} \\ &= \{(1, 1), (2, 1), (3, 1), (4, 1)\} \\ R^4 &= R^3 \circ R = \{(1, 1), (2, 1), (3, 1), (4, 1)\} \circ \{(1, 1), (2, 1), (3, 2), (4, 3)\} \\ &= \{(1, 1), (2, 1), (3, 1), (4, 1)\} \end{aligned}$$

Como $R^4 = R^3$ entonces $R^n = R^3$ para $n > 3$.

7.7. Representación de relaciones**7.7.1. Representación de relaciones utilizando matrices**

Una relación entre conjuntos finitos puede ser representada usando matrices de ceros y unos. Suponer que R es una relación del conjunto $A = \{a_1, a_2, a_3, \dots, a_m\}$ al con-

junto $B = \{b_1, b_2, b_3, \dots, b_n\}$. La relación R puede ser representada por la matriz $M_R = [m_{ij}]$, donde

$$m_{ij} = \begin{cases} 1 & \text{si } (a_i, b_j) \in R \\ 0 & \text{si } (a_i, b_j) \notin R \end{cases}$$

Ejemplo 18:

Sean los conjuntos $A = \{1, 2, 3\}$ y $B = \{1, 2\}$, sea la relación $R = \{(a, b) \mid a > b\} = \{(2, 1), (3, 1), (3, 2)\}$ definida del conjunto A al conjunto B . La representación de la relación R por medio de matrices es la siguiente:

$$M_R = \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 1 & 1 \end{bmatrix}$$

Ejemplo 19:

Sea la relación R en el conjunto $A = \{1, 2, 3, 4\}$. ¿Cuáles pares ordenados están en la relación R representados por la matriz

$$M_R = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{bmatrix} \quad ?$$

La relación contiene los siguientes pares ordenados:

$$R = \{(1, 2), (2, 1), (2, 3), (2, 4), (3, 2), (3, 3), (4, 2), (4, 4)\}.$$

Las matrices cuadradas son utilizadas para determinar si las relaciones tienen ciertas propiedades. Ellas son:

- La propiedad de reflexividad: si la diagonal principal esta compuesta de unos.

$$\begin{bmatrix} 1 & & & \\ & 1 & & \\ & & \ddots & \\ & & & 1 \\ & & & & 1 \end{bmatrix}$$

Recordar que R es reflexiva si únicamente si $(a_i, a_i) \in R$ para $i = 1, 2, 3, \dots, n$

■ La propiedad de Simetría.

$$\begin{bmatrix} & & 1 & \\ & \nearrow & & \\ 1 & & \ddots & \\ & & & \nearrow & 0 \\ & & 0 & & \end{bmatrix}$$

Recordar que R es simétrica si y únicamente si $(a, b) \in R$ si $M_r = (M_r)^t$ entonces R es simétrica

■ La propiedad de Antisimetría.

$$\begin{bmatrix} & & 1 & & \\ & \nearrow & & & 0 \\ 0 & & \nearrow & & 0 \\ & 0 & & \nearrow & \\ & & 1 & & \end{bmatrix}$$

Recordar que R es antisimétrica si y únicamente si $(a, b) \in R$ y $(b, a) \in R$ implica que $a = b$

Ejemplo 20:

Suponer que la relación R en un conjunto es representada por la matriz:

$$M_r = \begin{bmatrix} 1 & 1 & 0 \\ 1 & 1 & 1 \\ 0 & 1 & 1 \end{bmatrix}$$

La relación R es reflexiva.

La relación R es simétrica.

La relación R no es antisimétrica.

Las operaciones de unión e intersección de relaciones se pueden realizar con la representación en matrices

$$M_{R_1 \cup R_2} = M_{R_1} \vee M_{R_2}$$

$$M_{R_1 \cap R_2} = M_{R_1} \wedge M_{R_2}$$

Ejemplo 21:

Las relaciones R_1 y R_2 en un conjunto A son representados por las matrices

$$M_{R_1} = \begin{bmatrix} 1 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \quad \text{y} \quad M_{R_2} = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \\ 1 & 0 & 0 \end{bmatrix}$$

$$M_{R_1 \cup R_2} = M_{R_1} \vee M_{R_2} = \begin{bmatrix} 1 & 0 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 0 \end{bmatrix}$$

$$M_{R_1 \cap R_2} = M_{R_1} \wedge M_{R_2} = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

La composición de relaciones que son representadas por medio de matrices es $M_{S \circ R} = M_R \odot M_S$, donde \odot es la multiplicación de matrices binarias.

Ejemplo 22:

Encontrar la matriz que represente la relación $S \circ R$ donde las matrices que representa R y S son:

$$M_R = \begin{bmatrix} 1 & 0 & 1 \\ 1 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad \text{y} \quad M_S = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 1 \end{bmatrix}$$

$$M_{S \circ R} = M_R \odot M_S = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 1 & 1 \\ 0 & 0 & 0 \end{bmatrix}$$

La matriz que representa la composición de dos relaciones puede ser usada para encontrar la matriz de M_{R^n} . En particular $M_{R^n} = M_R^n$

Ejemplo 23:

Encontrar la matriz que representa la relación R^2 , donde la matriz que representa la relación R es:

$$M_R = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 1 & 1 \\ 1 & 0 & 0 \end{bmatrix}$$

El resultado es el siguiente:

$$M_{R^2} = M_R^2 = M_R \odot M_R = \begin{bmatrix} 0 & 1 & 1 \\ 1 & 1 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$

7.8. Relaciones n-arias y su uso computacional

Normalmente se establecen relaciones entre elementos de dos o más conjuntos. Por ejemplo, se presenta una relación entre el nombre profesor, título de profesional del profesor y nombre del programa académico en el cual trabaja el profesor.

Las relaciones entre elementos de dos o más conjuntos se conocen con el nombre de relación n -aria y son el fundamento matemático sobre el cual se apoya el modelo de bases de datos relacional propuesto por E. F. Codd en 1970 [C1970] y que actualmente se encuentra vigente en los sistemas manejadores de bases de datos.

Definición: Sean $A_1, A_2, A_3, \dots, A_n$ conjuntos. Una relación n -aria en estos conjuntos es un subconjunto de $A_1 \times A_2 \times A_3 \times \dots \times A_n$. Los conjuntos $A_1, A_2, A_3, \dots, A_n$ se llaman dominios de los atributos de la relación y n es el grado de la relación.

Ejemplo 24:

Sea R la relación “Horario de Clase” que consta de las 6-tuplas (o registros de seis atributos) (*Materia*, *Grupo*, *Día*, *HoraComienzo*, *HoraFinalización*, *Salón*). Por facilidad este tipo de relaciones se representan por medio de tablas. La representación tabular de la relación “Horario de Clase” y la forma de almacenar información en ella, es:

Materia	Grupo	Día	HoraComienzo	HoraFinalización	Salón
Matemáticas I	1	Lunes	07:00	09:00	L101
Lógica Matemática	3	Miércoles	16:00	18:00	Y309
Programación III	2	Viernes	14:00	16:00	R211

Cuadro 7.1: Relación HORARIO DE CLASE

7.9. El álgebra relacional en bases de datos

El álgebra relacional es un lenguaje de consulta. Consta de un conjunto de operaciones que toman como entrada una o dos relaciones y producen como resultado una nueva relación.

Las operaciones que se pueden realizar sobre las relaciones en el álgebra relacional son:

- Selección.
- Proyección.
- Unión.
- Intersección.

- Diferencia.
- Producto cartesiano.
- Asignación.

Las operaciones de selección y proyección se denominan operaciones unarias porque operan sobre una sola relación. El resto de las operaciones trabajan sobre pares de relaciones y se denominan, por lo tanto, operaciones binarias.

7.9.1. La operación de selección

La operación de selección sobre relaciones escoge las tuplas que satisfacen un predicado dado. Se utiliza la letra griega sigma minúscula (σ) para denotar la selección. El predicado aparece como subíndice de σ . La relación (tabla) del argumento se da entre paréntesis a continuación de σ .

Ejemplo 25:

Sea la relación PROFESOR:

Cédula	Nombre	Programa	CorreoElectrónico
8'522.870	Jovanny Castaño	Ing. Sistemas	isokmio@gmail.com
9'723.523	Juan Trujillo	Ing. Física	juanpatru@utp.edu.co
6'489.438	Ramiro Barros	Ing. Electrónica	ramiro@sirius.utp.edu.co
29'598.441	Paula Villa	Ing. Sistemas	paula2713@gmail.com
7'747.641	Darwin Lotero	Lic. Matemáticas	darwin@utp.edu.co
29'837.638	Ana María López	Ing. Sistemas	anamayi@utp.edu.co
30'857.753	Gabriela Bedoya	Lic. Matemáticas	gabybedo@hotmail.com
6'922.525	Hugo Morales	Ing. Sistemas	huhumor@utp.edu.co

Cuadro 7.2: Relación PROFESOR

Seleccionar las tuplas (registros) de la relación PROFESOR donde el programa sea Ing. Sistemas.

La consulta en álgebra relacional es:

$$\sigma_{\text{programa}=\text{"Ing. Sistemas"}}(\text{PROFESOR})$$

el resultado de la consulta es presentado en la tabla 7.3.

Se permiten las comparaciones que utilizan los operadores relacionales: $=$, \neq , $<$, \leq , $>$ o \geq en el predicado de la selección. Además, se pueden combinar varios predicados en uno mayor utilizando los conectivos lógicos \wedge y \vee .

Cédula	Nombre	Programa	CorreoElectrónico
8'522.870	Jovanny Castaño	Ing. Sistemas	isokmio@gmail.com
29'598.441	Paula Villa	Ing. Sistemas	paula2713@gmail.com
29'837.638	Ana María López	Ing. Sistemas	anamayi@utp.edu.co
6'922.525	Hugo Morales	Ing. Sistemas	huhumor@utp.edu.co

Cuadro 7.3: Profesores de Ingeniería de Sistemas

Ejemplo 26:

Seleccionar las tuplas de la relación PROFESOR donde el programa sea Ing. Sistemas o Ing. Electrónica.

La consulta en álgebra relacional es:

$$\sigma_{\text{programa}=\text{"Ing. Sistemas"} \vee \text{programa}=\text{"Ing. Electrónica"}} (\text{PROFESOR})$$

el resultado de la consulta es:

Cédula	Nombre	Programa	CorreoElectrónico
8'522.870	Jovanny Castaño	Ing. Sistemas	isokmio@gmail.com
6'489.438	Ramiro Barros	Ing. Electrónica	ramiro@sirius.utp.edu.co
29'598.441	Paula Villa	Ing. Sistemas	paula2713@gmail.com
29'837.638	Ana María López	Ing. Sistemas	anamayi@utp.edu.co
6'922.525	Hugo Morales	Ing. Sistemas	huhumor@utp.edu.co

Cuadro 7.4: Profesores de las Ingenierías de Sistemas o Electrónica

7.9.2. La operación proyección

Suponer que se desea hacer una lista de todos los nombres de los profesores, pero sin que aparezcan el resto de información en los otros atributos (es decir, sólo se quiere proyectar el atributo nombre de la relación PROFESOR).

La operación proyección es una operación unaria que devuelve su relación de argumentos, excluyendo algunos argumentos. Como las relaciones son conjuntos entonces se eliminan todas las tuplas (filas) duplicadas.

La proyección se denota por la letra griega mayúscula Pi (Π). Como subíndice de Π se coloca la lista de atributos que se desea proyectar en la relación de resultado. La relación (tabla) del argumento se da entre paréntesis a continuación de Π .

Ejemplo 27:

Listar el nombre de los profesores de la relación PROFESOR (ver tabla 7.2).

La consulta en álgebra relacional es:

$$\Pi_{nombre}(\text{PROFESOR})$$

donde se obtiene la siguiente relación como resultado:

Nombre
Jovanny Castaño
Juan Trujillo
Ramiro Barros
Paula Villa
Darwin Lotero
Ana María López
Gabriela Bedoya
Hugo Morales

Cuadro 7.5: Listado de nombres de los profesores

7.9.3. Composición de operaciones relacionales

Es importante que el resultado de una operación relacional sea también una relación. Considerar la consulta más compleja: “Listar el nombre y el correo electrónico de los estudiantes del programa de Ing. Sistemas”. Para esto, considerar que se tiene la relación ESTUDIANTE presentada en la tabla 7.6.

Código	Nombre	Programa	CorreoElectrónico
13'383.458	Sebastián Gómez	Ing. Sistemas	sebasgomez@utp.edu.co
12'137.638	Felipe Abad	Ing. Sistemas	falabad@utp.edu.co
8'522.870	Jovanny Castaño	Maestría en Ing. Sistemas	isokmio@gmail.com
29'598.441	Paula Villa	Maestría en Ing. Sistemas	paula2713@gmail.com
6'922.525	Hugo Morales	Maestría en Matemáticas	huhumor@utp.edu.co
12'549.937	Julian Moreno	Ing. Sistemas	junix@utp.edu.co
32'789.642	Claudia Montes	Ing. Eléctrica	claudia25@gmail.com
11'127.835	Andrés Munera	Ing. Física	fabicast@yahoo.com

Cuadro 7.6: Relación ESTUDIANTE

La consulta en álgebra relacional es la siguiente:

$$\Pi_{nombre, correoElectronico}(\sigma_{programa="Ing. Sistemas"}(\text{ESTUDIANTE}))$$

y el resultado que se genera es el presentado en la tabla 7.7.

Nombre	CorreoElectrónico
Sebastián Gómez	sebasgomez@utp.edu.co
Felipe Abad	falabad@utp.edu.co
Julian Moreno	junix@utp.edu.co

Cuadro 7.7: Listado de nombres y correos electrónicos de los estudiantes de Ing. Sistemas

El resultado de una operación del álgebra relacional es del mismo tipo (relación) que los datos de entrada, las operaciones del álgebra relacional pueden componerse para formar una expresión del álgebra relacional.

7.9.4. La operación de unión

Considerar que se quiere averiguar el nombre de todas las personas que son profesores, o estudiantes, o ambas cosas. Previamente ya se conoce la forma de averiguar los nombres de los profesores:

$$\Pi_{nombre}(\text{PROFESOR})$$

cuyo resultado es presentado en la tabla 7.5.

También se conoce la forma de averiguar los nombres de los estudiantes:

$$\Pi_{nombre}(\text{ESTUDIANTE})$$

cuyo resultado se presenta en la tabla 7.8.

Nombre
Sebastián Gómez
Felipe Abad
Jovanny Castaño
Paula Villa
Hugo Morales
Julian Moreno
Claudia Montes
Andrés Munera

Cuadro 7.8: Listado de los nombres de los estudiantes

Para contestar la pregunta hace falta la unión de estas dos relaciones, la cual se logra de la siguiente forma:

$$\Pi_{nombre}(\text{PROFESOR}) \cup \Pi_{nombre}(\text{ESTUDIANTE})$$

Nombre
Jovanny Castaño
Juan Trujillo
Ramiro Barros
Paula Villa
Darwin Lotero
Ana María López
Gabriela Bedoya
Hugo Morales

$$\cup$$

Nombre
Sebastián Gómez
Felipe Abad
Jovanny Castaño
Paula Villa
Hugo Morales
Julian Moreno
Claudia Montes
Andrés Munera

$$=$$

Nombre
Jovanny Castaño
Juan Trujillo
Ramiro Barros
Paula Villa
Darwin Lotero
Ana María López
Gabriela Bedoya
Hugo Morales
Sebastián Gómez
Felipe Abad
Julian Moreno
Claudia Montes
Andrés Munera

Se debe garantizar que la operación de unión se realice entre relaciones compatibles, en el tipo de atributos y cantidad de éstos. De esta forma, para que la operación de unión sea valida sobre las relaciones R y S se tienen que cumplir las siguientes dos condiciones:

1. Las relaciones R y S tienen que tener el mismo número de atributos.
2. El dominio del atributo i -ésimo de las relaciones R y S tiene que ser el mismo para todo i .

7.9.5. La operación de diferencia

La operación de diferencia de relaciones, denotada por el símbolo $-$, permite buscar las tuplas que están en una relación pero no en la otra. La expresión $R - S$ da como resultado una relación que contiene las tuplas que están en la relación R pero no en la relación S .

Ejemplo 28:

Listar los nombres de los profesores que no son estudiantes de la institución.

La consulta en álgebra relacional y el resultado es:

$\Pi_{nombre}(\text{PROFESOR})$	—	$\Pi_{nombre}(\text{ESTUDIANTE})$																			
<table><tr><th>Nombre</th></tr><tr><td>Jovanny Castaño</td></tr><tr><td>Juan Trujillo</td></tr><tr><td>Ramiro Barros</td></tr><tr><td>Paula Villa</td></tr><tr><td>Darwin Lotero</td></tr><tr><td>Ana María López</td></tr><tr><td>Gabriela Bedoya</td></tr><tr><td>Hugo Morales</td></tr></table>	Nombre	Jovanny Castaño	Juan Trujillo	Ramiro Barros	Paula Villa	Darwin Lotero	Ana María López	Gabriela Bedoya	Hugo Morales		<table><tr><th>Nombre</th></tr><tr><td>Sebastián Gómez</td></tr><tr><td>Felipe Abad</td></tr><tr><td>Jovanny Castaño</td></tr><tr><td>Paula Villa</td></tr><tr><td>Hugo Morales</td></tr><tr><td>Julian Moreno</td></tr><tr><td>Claudia Montes</td></tr><tr><td>Andrés Munera</td></tr></table>	Nombre	Sebastián Gómez	Felipe Abad	Jovanny Castaño	Paula Villa	Hugo Morales	Julian Moreno	Claudia Montes	Andrés Munera	
Nombre																					
Jovanny Castaño																					
Juan Trujillo																					
Ramiro Barros																					
Paula Villa																					
Darwin Lotero																					
Ana María López																					
Gabriela Bedoya																					
Hugo Morales																					
Nombre																					
Sebastián Gómez																					
Felipe Abad																					
Jovanny Castaño																					
Paula Villa																					
Hugo Morales																					
Julian Moreno																					
Claudia Montes																					
Andrés Munera																					
	—		<table><tr><th>Nombre</th></tr><tr><td>Juan Trujillo</td></tr><tr><td>Ramiro Barros</td></tr><tr><td>Darwin Lotero</td></tr><tr><td>Ana María López</td></tr><tr><td>Gabriela Bedoya</td></tr></table>	Nombre	Juan Trujillo	Ramiro Barros	Darwin Lotero	Ana María López	Gabriela Bedoya												
Nombre																					
Juan Trujillo																					
Ramiro Barros																					
Darwin Lotero																					
Ana María López																					
Gabriela Bedoya																					

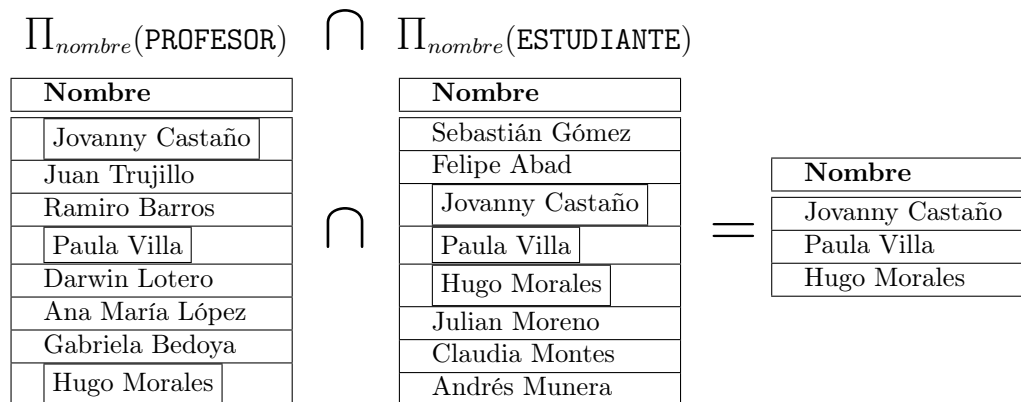
Como en el caso de la operación de unión, hay que asegurarse que la diferencia de relaciones se realice entre relaciones compatibles.

7.9.6. La operación de intersección

La operación de intersección de relaciones, denotada por el símbolo \cap , permite buscar las tuplas que están al mismo tiempo en las dos relaciones. La expresión $R \cap S$ da como resultado una relación que contiene las tuplas que están al mismo en las relaciones R y S .

Ejemplo 29:

Listar el nombre de los profesores que actualmente son estudiantes de algún programa académico de la institución.



La operación de intersección de relaciones puede construirse a partir de la operación de diferencia, de la siguiente forma:

$$R \cap S = R - (R - S)$$

7.9.7. La operación de asignación

En ocasiones resulta conveniente escribir una expresión del álgebra relacional por partes utilizando la operación de asignación a una variable de relación temporal. La operación de asignación denotada por \leftarrow , actúa de manera parecida a la asignación de los lenguajes de programación.

7.9.8. La operación producto cartesiano

La operación producto cartesiano, denotada por el símbolo \times , permite combinar información de cualquier par de relaciones. El producto cartesiano de las relaciones R y S es representado por $R \times S$.

Como el nombre de un atributo puede estar en ambas relaciones R y S , entonces hay que distinguirlos en el esquema del producto cartesiano, la forma tradicional de resolver este inconveniente es adicionar al nombre del atributo el nombre de la relación. Para atributos que sólo aparecen en una de las dos relaciones, se suele omitir el prefijo con el nombre de la relación.

Sean las relaciones **ARTÍCULO** y **PUBLICACIÓN** que serán utilizadas posteriormente:

Id	Nombre	Revista
1	Dispositivos móviles como lupa parlante	Scientia Et Technica
2	Espectro ensanchado por secuencia directa	Scientia Et Technica
3	Estándar DICOM en imágenes médicas	Scientia Et Technica
4	El algoritmo Heapinsert	Scientia Et Technica

Cuadro 7.9: Relación ARTÍCULO

IdEscrivor	IdArtículo
9'723.523	3
29'598.441	2
29'598.441	3
13'383.458	1
6'922.525	4

Cuadro 7.10: Relación PUBLICACIÓN

Ejemplo 30:

Listar los nombres de los autores y de los artículos escritos por estudiantes de Ing. Sistemas.

Por claridad se utilizará la operación de asignación en el desarrollo de este ejemplo.

$$T1 \leftarrow \sigma_{\text{programa}=\text{"Ing. Sistemas"}} (\text{ESTUDIANTE})$$

Código	Nombre	Programa	CorreoElectrónico
13'383.458	Sebastián Gómez	Ing. Sistemas	sebasgomez@utp.edu.co
12'137.638	Felipe Abad	Ing. Sistemas	falabad@utp.edu.co
12'549.937	Julian Moreno	Ing. Sistemas	junix@utp.edu.co

Cuadro 7.11: Relación temporal T1

$$T2 \leftarrow \Pi_{\text{código, nombre}}(T1)$$

Código	Nombre
13'383.458	Sebastián Gómez
12'137.638	Felipe Abad
12'549.937	Julian Moreno

Cuadro 7.12: Relación temporal T2

$$T3 \leftarrow T2 \times \text{PUBLICACIÓN}$$

Código	Nombre	IdEscritor	IdArtículo
13'383.458	Sebastián Gómez	9'723.523	3
13'383.458	Sebastián Gómez	29'598.441	2
13'383.458	Sebastián Gómez	29'598.441	3
13'383.458	Sebastián Gómez	13'383.458	1
13'383.458	Sebastián Gómez	6'922.525	4
12'137.638	Felipe Abad	9'723.523	3
12'137.638	Felipe Abad	29'598.441	2
12'137.638	Felipe Abad	29'598.441	3
12'137.638	Felipe Abad	13'383.458	1
12'137.638	Felipe Abad	6'922.525	4
12'549.937	Julian Moreno	9'723.523	3
12'549.937	Julian Moreno	29'598.441	2
12'549.937	Julian Moreno	29'598.441	3
12'549.937	Julian Moreno	13'383.458	1
12'549.937	Julian Moreno	6'922.525	4

Cuadro 7.13: Relación temporal T3

$$T4 \leftarrow \Pi_{\text{nombre AS nombreAutor, idArtículo}}(\sigma_{\text{código=idEscritor}}(T3))$$

NombreAutor	IdArtículo
Sebastián Gómez	1

Cuadro 7.14: Relación temporal T4

$$T5 \leftarrow \Pi_{id, nombre \text{ AS } nombreArticulo}(\text{ARTÍCULO})$$

Id	NombreArtículo
1	Dispositivos móviles como lupa parlante
2	Espectro ensanchado por secuencia directa
3	Estándar DICOM en imágenes médicas
4	El algoritmo Heapinsert

Cuadro 7.15: Relación temporal T5

$$T6 \leftarrow T4 \times T5$$

NombreAutor	IdArtículo	Id	NombreArtículo
Sebastián Gómez	1	1	Dispositivos móviles como lupa parlante
Sebastián Gómez	1	2	Espectro ensanchado por secuencia directa
Sebastián Gómez	1	3	Estándar DICOM en imágenes médicas
Sebastián Gómez	1	4	El algoritmo Heapinsert

Cuadro 7.16: Relación temporal T6

$$T7 \leftarrow \Pi_{nombreAutor, nombreArticulo}(\sigma_{idArticulo=id} (T6))$$

NombreAutor	NombreArtículo
Sebastián Gómez	Dispositivos móviles como lupa parlante

Cuadro 7.17: Relación temporal T7

El resultado de la consulta es el presentando en la relación temporal T7.

7.10. SQL el lenguaje de consulta de las bases de datos

La estructura básica de una consulta SQL (Structured Query Language - Lenguaje de Consulta Estructurado) consiste de tres cláusulas: **SELECT**, **FROM** y **WHERE**. La cláusula **SELECT** se usa para listar los atributos deseados del resultado de una consulta. En la cláusula **FROM** se escriben las relaciones que deben ser analizadas en la evaluación de la consulta. En la cláusula **WHERE** se coloca la condición que deben cumplir algunos de los atributos de las relaciones que aparecen en la cláusula **FROM**.

Una consulta típica en SQL tiene la forma:

```

SELECT    Atributo1, Atributo2, Atributo3, ..., Atributon
FROM      Relación1, Relación2, ..., Relaciónm
WHERE     Condición;

```

Si se omite la cláusula **WHERE** se asume que la condición es siempre verdadera y que por lo tanto se cumple por defecto.

SQL hace el producto cartesiano (también conocido con el nombre de producto cruz) entre las relaciones que han sido incluidas en la cláusula **FROM**, para luego, aplicar la selección con respecto a la lista de atributos de la cláusula **SELECT** a partir de las tuplas que cumplen la condición de la cláusula **WHERE**.

7.10.1. La cláusula SELECT

Como sucede también en álgebra relacional, el resultado de una consulta SQL es una relación.

Un primer ejemplo de una consulta en SQL es: “Listar el nombre de los profesores que se encuentran presentes en la relación **PROFESOR**” (ver tabla 7.2).

La consulta en álgebra relacional es:

```

SELECT    nombre
FROM      PROFESOR;

```

A diferencia del álgebra relacional, SQL permite duplicados en los resultados de una consulta. Por ejemplo si el profesor “Hugo Morales” se encuentra en varios registros en la relación **PROFESOR**, entonces será mostrado en el resultado de la consulta tantas veces como esté.

En los casos donde se quiera exigir la eliminación de registros duplicados, se tiene que agregar la instrucción **DISTINCT** después de la cláusula **SELECT**. En la consulta anterior, esto sería:

```

SELECT    DISTINCT nombre
FROM      PROFESOR;

```

SQL permite usar la palabra clave **ALL** para especificar explícitamente que no se eliminan los registros duplicados. De esta forma la consulta quedaría:

```

SELECT    ALL nombre
FROM      PROFESOR;

```

El símbolo ***** se puede usar en la cláusula **SELECT** para indicar que se listan todos los atributos que se encuentran presentes en las relaciones de la cláusula **FROM**.

La cláusula **SELECT** permite renombrar los nombres de los atributos en la relación

de resultado, de esta forma se puede renombrar el atributo “nombre” de la relación de resultado de la siguiente consulta por “nombreProfesor”:

```
SELECT    nombre AS nombreProfesor
FROM      PROFESOR;
```

7.10.2. La cláusula FROM

La cláusula FROM define por sí misma un producto cartesiano de las relaciones que aparecen en ella. Por ejemplo, la siguiente consulta genera el mismo resultado de la tabla 7.13:

```
SELECT    *
FROM      T2, PUBLICACIÓN;
```

7.10.3. La cláusula WHERE

En la cláusula WHERE se escribe la condición que deben cumplir la tuplas o registros del producto cartesiano generado entre las relaciones que se encuentran presentes en la cláusula FROM, de esta forma el resultado obtenido en la tabla 7.14 puede ser generado con la siguiente consulta SQL:

```
SELECT    nombre AS nombreAutor, idArticulo
FROM      T2, PUBLICACIÓN
WHERE     código = idEscritor;
```

SQL en la cláusula WHERE permite trabajar directamente con los nombres de los conectivos lógicos *and*, *or* y *not* en lugar de utilizar los símbolos matemáticos que los representan: \wedge , \vee y \neg . Con el uso de estos conectivos lógicos se permite construir condiciones cada vez más complejas. De forma similar como en álgebra relacional, las condiciones que deben cumplir los registros o tuplas pueden contener los operadores de comparación: $<$, $<=$, $>$, $>=$, $=$ y $<>$.

Ejemplo 31:

La siguiente consulta SQL permite: “Listar los nombres de los autores y de los artículos escritos por estudiantes de Ing. Sistemas”, generando exactamente el mismo resultado de la tabla 7.17:

```
SELECT    ESTUDIANTE.nombre AS nombreAutor, ARTÍCULO.nombre AS nombreArtículo
FROM      ESTUDIANTE, PUBLICACIÓN, ARTÍCULO
WHERE     ESTUDIANTE.código = PUBLICACIÓN.idEscritor  AND
          PUBLICACIÓN.idArtículo = ARTÍCULO.id  AND
          ESTUDIANTE.programa = ‘Ing. Sistemas’;
```


7.10.4. Operaciones sobre relaciones

Las operaciones de unión, intersección y diferencia de relaciones se pueden trabajar en consultas de SQL. Así como en el álgebra relacional, se necesita que las relaciones que participan en las operaciones sean compatibles; esto es, deben tener, la misma cantidad de atributos y el mismo nombre y tipo en el i -ésimo atributo de los resultados de las consultas SQL que se están trabajando.

7.10.5. La operación de unión

La operación de unión de relaciones se trabaja en SQL por medio de la cláusula UNION, la cual elimina los registros duplicados automáticamente. Si por el contrario, se desea conservar los registros duplicados, entonces se debe utilizar la cláusula UNION ALL.

Ejemplo 32:

Previamente en la operación de unión de álgebra relacional se trabajo la consulta: “Listar el nombre de todas las personas que son profesores, o estudiantes, o ambas cosas”, ahora, utilizando SQL, el mismo resultado se obtiene de la siguiente forma:

```
(SELECT    nombre
FROM      PROFESOR)
UNION
(SELECT    nombre
FROM      ESTUDIANTE);
```

7.10.6. La operación de diferencia

La operación de diferencia entre relaciones se representa en SQL por la cláusula EXCEPT.

Ejemplo 33:

El ejemplo que se trabajo en álgebra relacional para explicar la operación de diferencia de relaciones es: “Listar los nombres de los profesores que no son estudiantes de la institución”. Utilizando el lenguaje SQL la consulta que permite obtener el mismo resultado es:

```
(SELECT    nombre
FROM      PROFESOR)
EXCEPT
(SELECT    nombre
FROM      ESTUDIANTE);
```

7.10.7. La operación de intersección

La operación de intersección entre relaciones se representa en SQL por intermedio de la cláusula `INTERSECT`.

Ejemplo 34:

La consulta SQL que permite obtener el “listado de los nombres de los profesores que actualmente son estudiantes de algún programa académico de la institución”, es:

```
(SELECT      nombre
FROM        PROFESOR)
INTERSECT
(SELECT      nombre
FROM        ESTUDIANTE);
```

consulta que permite generar exactamente el mismo resultado al obtenido en el ejemplo de la operación de intersección de relaciones del álgebra relacional.

En las últimas secciones de éste capítulo se han presentado sólo unas pequeñas nociones del “maravilloso mundo mágico de las bases de datos” ... aquí no se presentado ni un 5 % del tema de álgebra relacional y del SQL, se invita al lector a profundizar en estos temas, una buena referencia bibliográfica para estos temas es el libro: Fundamentos de Bases de Datos de Silberschatz, Korth y Sudarshan [SKS2002].

7.11. Ejercicios

- Sean los conjuntos $A = \{2, 3, 4, 5, 6\}$ y $B = \{15, 16, 17, 18, 19, 20\}$, listar los elementos de la relación $R \subseteq A \times B$ donde aRb si a divide (exactamente) a b .
- Sea el conjunto $A = \{1, 2, 3, 4, 5, 6, 7\}$ sobre el cual está definida la relación $R = \{(a, b) \mid a - b \text{ es múltiplo de } 3\}$. Para la relación R contestar las siguientes preguntas
 - ¿Cuáles son los pares ordenados de $|A \times A|$ que pertenecen a la relación R ?
 - ¿La relación R tiene la propiedad de reflexividad?
 - ¿La relación R tiene la propiedad de simetría?
 - ¿La relación R tiene la propiedad de antisimetría?
 - ¿La relación R tiene la propiedad de transitividad?
- Sean A y B conjuntos con $|B| = 3$. Si hay 4096 relaciones de A a B , ¿cuál es el valor de $|A|$?
- Sean los conjuntos $A = \{1, 3, 5\}$ y $B = \{2, 4, 6, 8\}$, determinar
 - $A \times B$

- b.) $|A \times B|$
- c.) El número de relaciones de A a B
- d.) El número de relaciones binarias en A
- e.) El número de relaciones de A a B que contenga (1, 2) y (3, 4)
- f.) El número de relaciones de A a B que contenga exactamente cuatro pares ordenados
- g.) El número de relaciones binarias en A que contenga como mínimo cinco pares ordenados

5. Sean las siguientes relaciones definidas en el conjunto $A = \{1, 2\}$

$$\begin{aligned}
 R_1 &= \{\} \\
 R_2 &= \{(1, 1)\} \\
 R_3 &= \{(1, 2)\} \\
 R_4 &= \{(2, 1)\} \\
 R_5 &= \{(2, 2)\} \\
 R_6 &= \{(1, 1), (1, 2)\} \\
 R_7 &= \{(1, 1), (2, 1)\} \\
 R_8 &= \{(1, 1), (2, 2)\} \\
 R_9 &= \{(1, 2), (2, 1)\} \\
 R_{10} &= \{(1, 2), (2, 2)\} \\
 R_{11} &= \{(2, 1), (2, 2)\} \\
 R_{12} &= \{(1, 1), (1, 2), (2, 1)\} \\
 R_{13} &= \{(1, 1), (1, 2), (2, 2)\} \\
 R_{14} &= \{(1, 1), (2, 1), (2, 2)\} \\
 R_{15} &= \{(1, 2), (2, 1), (2, 2)\} \\
 R_{16} &= \{(1, 1), (1, 2), (2, 1), (2, 2)\}
 \end{aligned}$$

- a.) ¿Cuáles de las relaciones cumplen la propiedad de reflexividad?
- b.) ¿Cuáles de las relaciones cumplen la propiedad de simetría?
- c.) ¿Cuáles de las relaciones cumplen la propiedad de antisimetría?
- d.) ¿Cuáles de las relaciones cumplen la propiedad de transitividad?

6. Sea el conjunto $A = \{1, 2, 3\}$.

- a.) ¿Cuántas relaciones hay en el conjunto A?
- b.) ¿Cuántas relaciones hay en el conjunto A que sean reflexivas?
- c.) ¿Cuántas relaciones hay en el conjunto A que sean simétricas?
- d.) ¿Cuántas relaciones hay en el conjunto A que sean antisimétricas?
- e.) ¿Cuántas relaciones hay en el conjunto A que sean transitivas?

7. Sea R una relación del conjunto A al conjunto B . La relación inversa de B a A , denotada por R^{-1} , es el conjunto de los pares ordenados $\{(b, a) \mid (a, b) \in R\}$. La relación complemento \overline{R} es el conjunto de pares ordenados $\{(a, b) \mid (a, b) \text{ no pertenece a } R\}$. Calcular R^{-1} y \overline{R} para:
- a.) $R = \{(a, b) \mid a < b\}$ en el conjunto de los números enteros.
 - b.) $R = \{(a, b) \mid a \text{ divide a } b\}$ en el conjunto de los números enteros.
 - c.) R la relación en el conjunto de todos los departamentos de Colombia, la relación R consiste de todos los pares ordenados (a, b) donde el departamento a limita con el departamento b .
 - d.) $R = \{(a, b) \mid a + b \leq 3\}$ en el conjunto de los números enteros.
 - e.) $R = \{(1, 2), (1, 3), (1, 4), (1, 5), (2, 3), (2, 4), (2, 5), (3, 4), (3, 5), (4, 5)\}$, en el conjunto $A = \{1, 2, 3, 4, 5\}$.
 - f.) $R = \{(1, 1), (1, 2), (2, 1), (2, 2), (3, 3), (4, 4), (4, 5), (5, 4), (5, 5), (6, 6)\}$, en el conjunto $A = \{1, 2, 3, 4, 5, 6\}$
8. Sea R la relación en el conjunto $\{1, 2, 3, 4, 5\}$ que contiene los pares ordenados $(1, 2), (1, 3), (2, 3), (2, 4), (3, 1), (3, 4), (3, 5), (4, 2), (4, 5), (5, 1), (5, 2)$ y $(5, 4)$. Calcular:
- a.) R^2
 - b.) R^3
 - c.) R^4
 - d.) R^5

7.12. Preguntas tipo ECAES

1. Sean A y B conjuntos no vacíos, ¿cuántas relaciones hay de A a B ?¹
 - A.) $|A| \cdot |B|$
 - B.) $\binom{|A| \cdot |B|}{2}$
 - C.) $(|A \times B|)^2$
 - D.) $2^{|A|+|B|}$
 - E.) $2^{|A| \cdot |B|}$

2. Sean A y B conjuntos no vacíos, ¿cuántas relaciones hay que contengan exactamente tres pares ordenados de A a B ?
 - A.) 0
 - B.) $\binom{|A| \cdot |B|}{3}$
 - C.) $\binom{|A| + |B|}{3}$
 - D.) $\binom{\text{MAX}(|A|, |B|)}{3}$
 - E.) $(|A| \cdot |B|)^3$

3. Para $A = \{1, 2, 3, 4, 5, 6\}$, sea la siguiente relación $R = \{(1, 1), (1, 2), (2, 1), (2, 2), (3, 3), (4, 4), (4, 5), (5, 4), (5, 5), (6, 6)\}$ del conjunto A al conjunto A .
 ¿Cuál es la única respuesta correcta con respecto a la relación R ?
 - A.) R no es reflexiva, no es simétrica y no es transitiva.
 - B.) R es reflexiva, no es simétrica y es transitiva.
 - C.) R no es reflexiva, es simétrica y es transitiva.
 - D.) R es reflexiva, es simétrica y no es transitiva.
 - E.) R es reflexiva, es simétrica y es transitiva.

4. Sea $R = \{(a, b) \mid a - b \text{ es múltiplo de } 3\}$. Si R está definida sobre el conjunto $A = \{1, 2, 3, 4, 5, 6, 7\}$, ¿Cuál es el conjunto de pares ordenados que pertenecen a R ?
 - A.) $R = \{(1, 1), (2, 2), (3, 3), (4, 4), (5, 5), (6, 6), (7, 7)\}$

¹Las preguntas desde la 1 hasta la 8 son del autor.

B.) $R = \{(1, 4), (4, 1), (1, 7), (7, 1), (2, 5), (5, 2), (3, 6), (6, 3), (4, 7), (7, 4)\}$

C.) $R = \{(1, 1), (1, 4), (4, 1), (1, 7), (7, 1), (2, 2), (2, 5), (5, 2), (3, 3), (3, 6), (6, 3), (4, 4), (4, 7), (7, 4)\}$

D.) $R = \{(1, 1), (1, 4), (4, 1), (1, 7), (7, 1), (2, 2), (2, 5), (5, 2), (3, 3), (3, 6), (6, 3), (4, 4), (4, 7), (7, 4), (5, 5), (6, 6), (7, 7)\}$

E.) $R = \{\}$

5. Una relación R es llamada asimétrica si $(a, b) \in R$ implica que $(b, a) \notin R$. Sean las siguientes relaciones en el conjunto de todas las personas del mundo:

- $R_1 = \{(a, b) \mid a \text{ y } b \text{ tienen la misma edad}\}$
- $R_2 = \{(a, b) \mid a \text{ y } b \text{ tienen los mismos padres}\}$
- $R_3 = \{(a, b) \mid a \text{ y } b \text{ tienen solo un padre en común}\}$
- $R_4 = \{(a, b) \mid a \text{ y } b \text{ se conocen}\}$

¿Cuál de las relaciones anteriores es asimétrica?

- A.) Únicamente R_1
 B.) Únicamente R_2
 C.) Únicamente R_3
 D.) Únicamente R_4
 E.) Ninguna relación

6. La relación complemento denotada por \overline{R} es el conjunto de pares ordenados $\{(a, b) \mid (a, b) \notin R\}$. ¿Cuál es la relación complemento \overline{R} de $R = \{(a, b), (a, c), (b, c)\}$ en el conjunto $A = \{a, b, c\}$?

- A.) $\overline{R} = \{(b, a), (c, a), (c, b)\}$
 B.) $\overline{R} = \{(a, a), (b, b), (c, c)\}$
 C.) $\overline{R} = \{(a, a), (a, b), (b, b), (a, c), (b, c), (c, c)\}$
 D.) $\overline{R} = \{(a, a), (b, a), (b, b), (c, a), (c, b), (c, c)\}$

7. Sea la relación $R = \{(1, 1), (2, 2), (3, 3), (4, 4), (5, 5), (1, 3), (3, 1), (1, 5), (5, 1), (2, 4), (4, 2), (3, 5), (5, 3)\}$, que está definida sobre el conjunto $A = \{1, 2, 3, 4, 5\}$.

¿Qué es $R \circ R$?

- A.) $R \circ R = R$

B.) $R \circ R = \{(1, 3), (3, 1), (1, 5), (5, 1), (2, 4), (4, 2), (3, 5), (5, 3)\}$

C.) $R \circ R = \{(1, 1), (2, 2), (3, 3), (4, 4), (5, 5)\}$

D.) $R \circ R = \{(1, 1), (2, 2), (3, 3), (4, 4), (5, 5), (1, 3), (1, 5), (2, 4), (3, 5)\}$

E.) $R \circ R = \{(1, 1), (2, 2), (3, 3), (4, 4), (5, 5), (3, 1), (5, 1), (4, 2), (5, 3)\}$

8. Sea la relación $R = \{(a, b) \mid a > b\}$, que está definida sobre el conjunto de los enteros.

¿Qué es $R \circ R$?

A.) $R \circ R = R$

B.) $R \circ R = \{(a, b) \mid a > b + 1\}$

C.) $R \circ R = \{(a, b) \mid a \geq b\}$

D.) $R \circ R = \{(a, b) \mid a + 1 > b\}$

E.) $R \circ R = \{\}$

Capítulo 8

Relaciones de equivalencia

Definición (Relación de Equivalencia):

Una relación en un conjunto A es llamada una relación de equivalencia si ésta es reflexiva, simétrica y transitiva.

Dos elementos que son relacionados por una relación de equivalencia son llamados equivalentes.

Ejemplo 1:

Supóngase que R es una relación en el conjunto de las cadenas escritas con el alfabeto español tales que aRb si y únicamente si $l(a) = l(b)$ donde $l(x)$ es la longitud de la cadena x . ¿Es R una relación de equivalencia?

Para que R sea una relación de equivalencia ésta tiene que ser reflexiva, simétrica y transitiva.

Reflexiva: $l(a) = l(a)$, de ésta forma aRa , por lo tanto R es reflexiva.

Simétrica: aRb lo que indica que $l(a) = l(b)$ como $l(b) = l(a)$ esto indica que bRa por lo tanto R es simétrica.

Transitiva: suponer que aRb y bRc , entonces $l(a) = l(b)$ y $l(b) = l(c)$, donde se obtiene que $l(a) = l(c)$, por lo tanto R es transitiva.

Como la relación R es reflexiva, simétrica y transitiva, entonces es una relación de equivalencia.

Ejemplo 2:

Sea R la relación en el conjunto de los números reales tales que aRb si y solo si, $a - b$ es un entero. ¿Es R una relación de equivalencia?

Para que R sea una relación de equivalencia ésta tiene que ser reflexiva, simétrica y transitiva.

Reflexiva: $a - a = 0$ que es un número entero para todos los números reales a . Así aRa . La relación R si es reflexiva.

Simétrica: Suponer que aRb , entonces $a - b$ es un número entero, también se tiene que $b - a$ es un número entero y se obtiene entonces bRa , de ésta forma se obtiene la propiedad de simetría en la relación R .

Transitiva : si aRb y bRc , entonces $a - b$ y $b - c$ son enteros, además $a - c = (a - b) + (b - c)$ que es también un entero, lo que conduce a tener aRc . Por lo tanto R es transitiva.

Como la relación R es reflexiva, simétrica y transitiva, entonces R es una relación de equivalencia.

Ejemplo 3: (Congruencia módulo m)

Sea m un entero positivo con $m > 1$, mostrar que la relación $R = \{(a, b) \mid a \equiv b \pmod{m}\}$ es una relación de equivalencia en el conjunto de los números enteros.

Recordar que $a \equiv b \pmod{m}$ si y solo si m divide exactamente a $a - b$.

Reflexiva: aRa , $a \equiv a \pmod{m}$, $a - a = 0$, cero es divisible exactamente por m por lo tanto la relación R es reflexiva.

Simétrica: Ahora suponer que aRb , $a \equiv b \pmod{m}$, $a - b = k \cdot m$, de este modo $a - b$ es divisible por m . entonces $b - a = (-k) \cdot m$, por lo tanto bRa ($b \equiv a \pmod{m}$). De ésta forma la relación de congruencia módulo m es simétrica.

Transitividad: Si aRb y bRc es porque $a \equiv b \pmod{m}$ y $b \equiv c \pmod{m}$, por lo tanto el número entero m divide exactamente a $a - b$ y a $b - c$, de esta forma:

$$a - b = k \cdot m \quad \text{y} \quad b - c = l \cdot m, \quad \text{donde } k, l \in \mathbb{Z}$$

$$\begin{aligned} a - c &= a - c - b + b \\ &= (a - b) + (b - c) \\ &= k \cdot m + l \cdot m \\ &= (k + l) \cdot m \\ &= j \cdot m, \quad \text{donde } j = k + l, j \in \mathbb{Z}. \end{aligned}$$

como $a - c = j \cdot m$ entonces se cumple que $a \equiv c \pmod{m}$, de ésta forma se cumple que la relación R tiene la propiedad de transitividad.

Como la relación R es reflexiva, simétrica y transitiva entonces R es una relación de equivalencia.

8.1. Clases de equivalencia

Definición (Clase de equivalencia):

Sea R una relación de equivalencia en un conjunto A . El conjunto de todos los elementos que están relacionados con el elemento a (para $a \in A$) es llamada la clase de equivalencia de a . La clase de equivalencia de a con respecto a la relación de equivalencia R es denotada con $[a]_R$.

En otras palabras, si R es una relación de equivalencia en un conjunto A , la clase de equivalencia del elemento a es:

$$[a]_R = \{s \mid (a, s) \in R\}$$

Si $b \in [a]_R$, entonces b es llamado representante de ésta clase de equivalencia. Cualquier elemento de una clase de equivalencia puede representarla.

Ejemplo 4:

¿Cuál es la clase de equivalencia para el número 0 en la relación $a \equiv b \pmod{4}$?

La clase de equivalencia del 0 contiene todos los números enteros a tales que $a \equiv 0 \pmod{4}$, es decir, contiene todos los números enteros a que son divisibles de forma exacta por 4.

$$[0] = \{\dots, -8, -4, 0, 4, 8, \dots\}$$

8.2. Clases de equivalencia y particiones

Teorema:

Sea R una relación de equivalencia en un conjunto A . Los siguientes tres ítems son equivalentes:

1. aRb
2. $[a]_R = [b]_R$
3. $[a]_R \cap [b]_R \neq \emptyset$

Sea R una relación de equivalencia en un conjunto A . La unión de todas las clases de equivalencia de R da como resultado el conjunto A

$$\bigcup_{a \in A} [a]_R = A$$

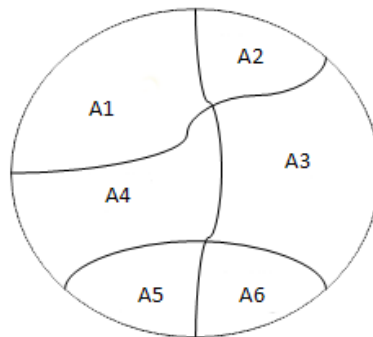
Si $[a]_R \cap [b]_R = \emptyset$ entonces $[a]_R \neq [b]_R$.

Las observaciones anteriores muestran que las clases de equivalencia forman una *partición* del conjunto A , donde ellas fraccionan al conjunto A en subconjuntos disyuntos. Más precisamente, una partición de un conjunto S es una colección de subconjuntos no vacíos disyuntos de S que tienen a S como su unión. En otras palabras, la colección de subconjuntos A_i , $i \in I$ (donde I es el conjunto de índices) forma una partición de conjunto S si y solo si $A_i \neq \emptyset$ para $i \in I$, $A_i \cap A_j = \emptyset$ cuando $i \neq j$, y $\bigcup_{i \in I} A_i = S$

Ejemplo 5:

Sea el conjunto $S = \{1, 2, 3, 4, 5, 6\}$. La colección de conjuntos $A_1 = \{1, 2, 3\}$, $A_2 = \{4, 5\}$ y $A_3 = \{6\}$ forman una partición del conjunto S , donde estos conjuntos son disyuntos y su unión es el conjunto S .

Los subconjuntos en una partición son clases de equivalencia. Dos elementos son equivalentes con respecto a la relación si y únicamente si ellos están en el mismo subconjunto de la partición.



Teorema:

Sea R una relación de equivalencia en un conjunto S . Entonces las clases de equivalencia de R forman una partición de S . Así mismo, dada una partición $\{A_i \mid i \in I\}$ del conjunto S , hay una relación de equivalencia R que tiene el conjunto A_i , $i \in I$, como clase de equivalencia.

Ejemplo 6:

Listar los pares ordenados en la relación de equivalencia R producidos por la partición $A_1 = \{1, 2, 3\}$, $A_2 = \{4, 5\}$ y $A_3 = \{6\}$ del conjunto $S = \{1, 2, 3, 4, 5, 6\}$

Los subconjuntos en la partición son clases de equivalencia de R , donde el par ordenado $(a, b) \in R$ si y solo si a y b están en el mismo subconjunto de la partición.

- Pares ordenados que aparecen en la relación de equivalencia R gracias a los elementos de la partición (o clase de equivalencia) A_1 : $(1, 1)$, $(1, 2)$, $(1, 3)$, $(2, 1)$, $(2, 2)$, $(2, 3)$, $(3, 1)$, $(3, 2)$ y $(3, 3)$.
- Pares ordenados que aparecen en la relación de equivalencia R gracias a los elementos de la partición (o clase de equivalencia) A_2 : $(4, 4)$, $(4, 5)$, $(5, 4)$ y $(5, 5)$
- Pares ordenados que aparecen en la relación de equivalencia R gracias a los elementos de la partición (o clase de equivalencia) A_3 : $(6, 6)$

De esta forma la relación de equivalencia R contiene únicamente los siguientes pares ordenados:

$$R = \{(1, 1), (1, 2), (1, 3), (2, 1), (2, 2), (2, 3), (3, 1), (3, 2), (3, 3), (4, 4), (4, 5), (5, 4), (5, 5), (6, 6)\}$$

Hay m diferentes clases de congruencia módulo m , correspondientes a los m diferentes residuos posibles cuando un entero es dividido por m . Estas m clases de congruencia son denotadas por $[0]_m$, $[1]_m$, \dots , $[m-1]_m$. Ellos forman una partición del conjunto de los enteros.

Ejemplo 7:

¿Cuales son los conjuntos en la partición de los números enteros para la relación $a \equiv b \pmod{4}$?

Hay cuatro clases de congruencia correspondientes a $[0] \equiv \pmod{4}$, $[1] \equiv \pmod{4}$, $[2] \equiv \pmod{4}$ y $[3] \equiv \pmod{4}$, ellas son los conjuntos:

$$[0] \equiv \pmod{4} = \{\dots, -8, -4, 0, 4, 8, \dots\}$$

$$[1] \equiv \pmod{4} = \{\dots, -7, -3, 1, 5, 9, \dots\}$$

$$[2] \equiv \pmod{4} = \{\dots, -6, -2, 2, 6, 10, \dots\}$$

$$[3] \equiv \pmod{4} = \{\dots, -5, -1, 3, 7, 11, \dots\}$$

Las clases de congruencia son disyuntas, y cada número entero está exactamente una sola de ellas.

8.3. Conjuntos parcialmente ordenados

Una relación $R \subseteq S \times S$ genera un ordenamiento parcial en el conjunto S si ésta cumple las propiedades de reflexividad, antisimetría y transitividad. Un conjunto S que es parcialmente ordenado con respecto a la relación R es denotado por (S, R) .

Ejemplo 8:

¿La relación “mayor o igual que” (\geq) genera un ordenamiento parcial en el conjunto de los números enteros?

Para dar respuesta a este interrogante primero se deben analizar las propiedades de reflexividad, antisimetría y transitividad de la relación, para esto se tiene:

Reflexiva: Sea $a \in \mathbb{Z}$, como cualquier número entero es mayor o igual a él mismo ($a \geq a$) entonces aRa , de donde se concluye que la relación R tiene la propiedad de reflexividad.

Antisimétrica: Sea $a, b \in \mathbb{Z}$, si aRb es porque $(a \geq b)$ y si bRa es porque $(b \geq a)$, la única forma de que se presente al mismo tiempo aRb y bRa es cuando $a = b$, con lo cual se cumple la propiedad de antisimetría en la relación R .

Transitiva: Sea $a, b, c \in \mathbb{Z}$, si aRb y bRc es porque $(a \geq b)$ y $(b \geq c)$ de donde se obtiene que $(a \geq c)$ con lo cual se cumple que aRc , con lo cual se concluye que la relación R cumple la propiedad de transitividad.

Como la relación “mayor o igual que” cumple las propiedades reflexiva, antisimétrica y transitiva entonces la relación genera un ordenamiento parcial en el conjunto de los números enteros, o expresado de una forma equivalente (\mathbb{Z}, \geq) es un conjunto parcialmente ordenado.

Ejemplo 9:

¿La relación divisibilidad ($|$) genera un ordenamiento parcial en el conjunto de los números enteros positivos?

Sea R la relación de divisibilidad, $R = \{(a, b) \mid a, b, m \in \mathbb{Z}^+ \wedge b = m \cdot a\}$, en palabras, aRb o $(a, b) \in R$ si a divide de forma exacta a b . La relación de divisibilidad R genera un ordenamiento parcial en el conjunto de los números enteros positivos si cumple las propiedades de reflexividad, antisimetría y transitividad. En el análisis de estas propiedades se tiene:

Reflexiva: Sea el número $a \in \mathbb{Z}^+$, a divide de forma exacta a a y se obtiene como resultado el número entero 1, sea cual sea el número entero positivo a , aRa , por lo tanto la relación de divisibilidad cumple la propiedad de reflexividad.

Antisimétrica: Sean los números $a, b \in \mathbb{Z}^+$, si aRb es porque a divide de forma exacta a b y si bRa es porque b divide de forma exacta a a , la única forma de que se presente al mismo tiempo aRb y bRa es cuando $a = b$, con lo cual se cumple la propiedad de antisimetría en la relación de divisibilidad.

Transitiva: Sean los números $a, b, c \in \mathbb{Z}^+$, si aRb es porque $b = m \cdot a$ para $m \in \mathbb{Z}^+$ y si bRc es porque $c = n \cdot b$ para $n \in \mathbb{Z}^+$, como $b = m \cdot a$ entonces $c = n \cdot (m \cdot a) = (n \cdot m) \cdot a = p \cdot a$ para $p = n \cdot m$ donde $p \in \mathbb{Z}^+$, como $c = p \cdot a$ entonces a divide de forma exacta a c , es decir aRc , de esta forma la relación de divisibilidad cumple la propiedad de transitividad.

Como la relación de “divisibilidad” cumple las propiedades reflexiva, antisimétrica y transitiva entonces la relación genera un ordenamiento parcial en el conjunto de los números enteros positivos, o expresado de una forma equivalente $(\mathbb{Z}^+, |)$ es un conjunto parcialmente ordenado.

Ejemplo 10:

Mostrar que la relación de inclusión (\subseteq) genera un ordenamiento parcial en el conjunto potencia de un conjunto S .

Recordar que el conjunto potencia de un conjunto S , es el conjunto que contiene todos los subconjuntos que se encuentran presentes en el conjunto S , la notación que se utiliza para representar dicho conjunto es $\mathcal{P}(S)$.

Sea $R = \{(A, B) \mid A, B \in \mathcal{P}(S) \wedge A \subseteq B\}$ para S que es un conjunto, la relación R es de orden parcial si cumple las propiedades de reflexividad, antisimetría y transitividad. En el análisis de las propiedades se tiene:

Reflexiva: Sea el conjunto $A \in \mathcal{P}(S)$, $A \subseteq A$, el conjunto A esta contenido en el conjunto A , por lo tanto se cumple la propiedad de reflexividad donde $(A, A) \in R$.

Antisimétrica: Sean los conjuntos $A, B \in \mathcal{P}(S)$, si $(A, B) \in R$ es porque $A \subseteq B$ y si $(B, A) \in R$ es porque $B \subseteq A$, la única posibilidad de que se presente al mismo tiempo $(A, B) \in R$ y $(B, A) \in R$ es cuando $A = B$, con lo cual la relación R cumple la propiedad de antisimetría.

Transitiva: Sean los conjuntos $A, B, C \in \mathcal{P}(S)$, si $(A, B) \in R$ y $(B, C) \in R$ es porque $A \subseteq B$ y $B \subseteq C$ de donde se deduce que $A \subseteq C$ lo que garantiza que $(A, C) \in R$, con lo cual la relación R cumple la propiedad de transitividad.

Como la relación de inclusión cumple las propiedades de reflexividad, antisimetría y transitividad, entonces el conjunto potencia es un conjunto parcialmente ordenado con respecto a la relación de inclusión, o expresado de una forma equivalente $(\mathcal{P}(S), \subseteq)$ es un conjunto parcialmente ordenado.

Definición:

Sea una relación $R \subseteq S \times S$, si la relación R genera un orden parcial sobre el conjunto S , para $a, b \in S$ se dice que los elementos a y b son comparables si $(a, b) \in R$ o $(b, a) \in R$. Cuando $(a, b) \notin R$ y $(b, a) \notin R$ se dice que los elementos a y b son incomparables.

Ejemplo 11:

En un ejemplo anterior ya se demostró que la relación de divisibilidad genera un orden parcial sobre el conjunto de los números enteros positivos, lo que es representado como $(\mathbb{Z}^+, |)$. ¿Son los números 2 y 4 comparables en $(\mathbb{Z}^+, |)$?, ¿son los números 5 y 2 comparables en $(\mathbb{Z}^+, |)$?

Para dar respuesta a las preguntas se tiene:

- Se cumple que $2, 4 \in \mathbb{Z}^+$ y que $(2, 4) \in R$ porque el número 2 divide de forma exacta al número 4, por este motivo los números 2 y 4 son comparables sobre $(\mathbb{Z}^+, |)$.
- Se cumple que $5, 2 \in \mathbb{Z}^+$, pero, $(5, 2) \notin R$ porque el número 5 no divide de forma exacta al número 2 y $(2, 5) \notin R$ porque el número 2 no divide de forma exacta al número 5, como $(5, 2) \notin R$ y $(2, 5) \notin R$ entonces los números 2 y 5 son incomparables sobre $(\mathbb{Z}^+, |)$.

Definición:

Sea una relación $R \subseteq S \times S$. La relación R genera un orden total en un conjunto S cuando ésta cumple las propiedades de reflexividad, antisimetría y transitividad, y adicionalmente, para cualquier par de elementos $a, b \in S$, a y b tienen que ser comparables.

En esta definición se exige que para que una relación R genere un orden total sobre un conjunto S , primero, la relación tiene que generar un orden parcial sobre el conjunto S , y luego se tiene que cumplir que cualquier par de elementos a y b del conjunto S sean comparables.

Ejemplo 12:

¿La relación de divisibilidad genera un orden total en el conjunto de los números enteros positivos?

En un ejemplo anterior ya se demostró que la relación de divisibilidad genera un orden parcial sobre \mathbb{Z}^+ , pero, no genera un orden total sobre \mathbb{Z}^+ porque existen pares ordenados de $\mathbb{Z}^+ \times \mathbb{Z}^+$ que son incomparables, por ejemplo los números 3 y 10 son incomparables.

Ejemplo 13:

¿La relación “mayor o igual que” genera un orden total en el conjunto de los números enteros?

En un ejemplo anterior ya se demostró que la relación “mayor o igual que” genera un orden parcial sobre \mathbb{Z} , ahora lo que hace falta es analizar si para cualquier par de números $a, b \in \mathbb{Z}$, a y b son comparables, para esto obligatoriamente se tiene que cumplir que $a \geq b$ o $b \geq a$, donde aRb o bRa con lo cual se cumple que cualquier par de números enteros a y b son comparables, por lo tanto la relación “mayor o igual que” genera un orden total sobre el conjunto de los números enteros.

8.4. Ejercicios

1. Demostrar o refutar que la relación $R = \{(a, b) \mid a - b \text{ es par}\}$ es una relación de equivalencia en el conjunto de los números enteros.
2. Demostrar o refutar que la relación $R = \{(a, b) \mid a \neq b\}$ es una relación de equivalencia en el conjunto de los números enteros.
3. Sea R una relación de equivalencia, ¿Qué se obtiene como resultado de $R \circ R$?
4. Sea la siguiente relación en el conjunto de todas las personas del mundo, $R = \{(a, b) \mid a \text{ y } b \text{ tienen el mismo año de nacimiento}\}$, ¿es R una relación de equivalencia?, si R es una relación de equivalencia, ¿Cuáles y cuántas son las clases de equivalencia?
5. Sea la siguiente relación en el conjunto de todas las personas del mundo, $R = \{(a, b) \mid a \text{ y } b \text{ tienen la misma fecha de cumpleaños}\}$, ¿es R una relación de equivalencia?, si R es una relación de equivalencia, ¿Cuáles y cuántas son las clases de equivalencia?

Nota: El hecho de que dos personas tengan la misma fecha de cumpleaños no indica que las dos personas tengan la misma fecha de nacimiento, por ejemplo Carlos y Manuel cumplen años el 22 de Julio, pero la fecha de nacimiento de Carlos es el 22 de Julio de 1968 mientras la fecha de nacimiento de Manuel es el 22 de Julio de 1975.

6. Sea la siguiente relación en el conjunto de todas las personas del mundo, $R = \{(a, b) \mid a \text{ y } b \text{ tienen al menos un mismo padre en común}\}$, ¿es R una relación de equivalencia?, si R es una relación de equivalencia, ¿Cuáles y cuántas son las clases de equivalencia?
7. Sea la siguiente relación en el conjunto de todas las personas del mundo, $R = \{(a, b) \mid a \text{ y } b \text{ hablan un lenguaje en común}\}$, ¿es R una relación de equivalencia?, si R es una relación de equivalencia, ¿Cuáles y cuántas son las clases de equivalencia?
8. Demostrar o refutar que la relación R en \mathbb{Z}^+ es un relación de equivalencia, donde R es definida por $a R b$ si y únicamente si $\tau(a) = \tau(b)$, donde $\tau(a)$ es igual al número de divisores positivos de a . Por ejemplo $2R_3$ y $4R_{25}$.

9. Demostrar o refutar que la relación R en \mathbb{N} es un relación de equivalencia, donde R es definida por $a R b$ si y únicamente si $\tau(a) = \tau(b)$, donde $\tau(a)$ es igual al número de cifras del número a . Por ejemplo $2R_3$ y $52R_{25}$.
10. Demostrar o refutar que la relación R en \mathbb{Z}^+ es un relación de equivalencia, donde R es definida por $a R b$ si y únicamente si $\tau(a) = \tau(b)$, donde $\tau(a)$ es igual al número de dígitos diferentes que conforman al número a . Por ejemplo $\tau(100) = 2$, $\tau(123) = 3$, $\tau(1020) = 3$ y $\tau(10000) = 2$, por lo tanto $123R_{1020}$ y $100R_{10000}$.
11. Demostrar o refutar que la relación R en \mathbb{N} es un relación de equivalencia, donde R es definida por $a R b$ si y únicamente si $\tau(a) = \tau(b)$, donde $\tau(a)$ devuelve el dígito menos significativo del número a . Por ejemplo $\tau(522) = 2$, $\tau(43) = 3$ y $\tau(7) = 7$, de esta forma, $52R_{112}$ y $1000R_{30}$.
12. Demostrar o refutar que la relación R en \mathbb{N} es un relación de equivalencia, donde R es definida por $a R b$ si y únicamente si $\tau(a) = \tau(b)$, donde $\tau(a)$ devuelve la suma de las cifras que componen al número a . Por ejemplo $\tau(522) = 9$, $\tau(43) = 7$ y $\tau(7) = 7$, de esta forma, $52R_{142}$ y $1000R_1$.
13. Sea R la relación en el conjunto de pares ordenados de enteros positivos tales que $((a, b), (c, d)) \in R$ si y únicamente si $ad = bc$. ¿Es R una relación de equivalencia?, si es así, ¿cuál es la clase de equivalencia de $(1, 2)$?
14. Sea R la relación en el conjunto de pares ordenados de enteros positivos tales que $((a, b), (c, d)) \in R$ si y únicamente si $a + d = b + c$. ¿Es R una relación de equivalencia?, si es así, ¿cuál es la clase de equivalencia de $(1, 2)$?
15. Sea R la relación en el conjunto de pares ordenados de enteros positivos tales que $((a, b), (c, d)) \in R$ si y únicamente si $a - d = b - c$. ¿Es R una relación de equivalencia?, si es así, ¿cuál es la clase de equivalencia de $(1, 2)$?
16. Sea R la relación en el conjunto de pares ordenados de enteros positivos tales que $((a, b), (c, d)) \in R$ si y únicamente si $\frac{a}{d} = \frac{c}{b}$. ¿Es R una relación de equivalencia?, si es así, ¿cuál es la clase de equivalencia de $(2, 5)$?
17. Sea R la relación en el conjunto de pares ordenados de enteros positivos tales que $((a, b), (c, d)) \in R$ si y únicamente si $a = c$. ¿Es R una relación de equivalencia?, si es así, ¿cuál es la clase de equivalencia de $(1, 4)$?, y, ¿cómo se representada dicha clase de equivalencia en el plano cartesiano?
18. Demostrar o refutar que la relación $R = \{(a, b) \mid a > b\}$ genera un ordenamiento parcial en el conjunto de los números enteros.
19. Demostrar o refutar que la relación $R = \{(a, b) \mid a < b\}$ genera un ordenamiento parcial en el conjunto de los números enteros.
20. Demostrar o refutar que la relación $R = \{(a, b) \mid a \leq b\}$ genera un ordenamiento parcial en el conjunto de los números enteros.

8.5. Preguntas tipo ECAES

1. Una forma de representar relaciones es por medio de matrices¹, $M[x][y] = 1$ si $(x, y) \in R$ y $M[x][y] = 0$ si $(x, y) \notin R$. Según esta definición, ¿cuál de las siguientes matrices representa una relación de equivalencia?

A.) $\begin{pmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{pmatrix}$

B.) $\begin{pmatrix} 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{pmatrix}$

C.) $\begin{pmatrix} 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 \end{pmatrix}$

D.) $\begin{pmatrix} 1 & 1 & 0 & 1 \\ 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 \end{pmatrix}$

E.) $\begin{pmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \end{pmatrix}$

2. Una relación en un conjunto A es llamada una Relación de Equivalencia si ésta es reflexiva, simétrica y transitiva. Sean las siguientes relaciones en el conjunto de todas las personas del mundo² :

- $R_1 = \{(a, b) \mid a \text{ y } b \text{ tienen la misma edad}\}$
- $R_2 = \{(a, b) \mid a \text{ y } b \text{ tienen los mismos padres}\}$
- $R_3 = \{(a, b) \mid a \text{ y } b \text{ tienen solo un padre en común}\}$
- $R_4 = \{(a, b) \mid a \text{ y } b \text{ se conocen}\}$

¿Cuáles relaciones anteriores son relaciones de equivalencia?

A.) Únicamente R_1 y R_2

B.) Únicamente R_2 y R_3

¹Pregunta reescrita a partir de una pregunta no liberada del ECAES de Ingeniería de Sistemas del año 2007.

²Las preguntas desde la 2 hasta la 5 son del autor.

C.) Únicamente R_3 y R_4

D.) Únicamente R_1 y R_4

3. Sean las siguientes relaciones en el conjunto de los números enteros:

- $R_1 = \{(a, b) \mid a - b \text{ es par}\}$
- $R_2 = \{(a, b) \mid a - b \text{ es impar}\}$
- $R_3 = \{(a, b) \mid a - b \text{ es múltiplo de } 3\}$
- $R_4 = \{(a, b) \mid a = 3b\}$

¿Cuáles de las relaciones anteriores son relaciones de equivalencia?

A.) Únicamente R_1 y R_2

B.) Únicamente R_2 y R_3

C.) Únicamente R_3 y R_4

D.) Únicamente R_2 y R_4

E.) Únicamente R_1 y R_3

4. Para $A = \{1, 2, 3, 4, 5, 6\}$, $R = \{(1, 1), (1, 2), (2, 1), (2, 2), (3, 3), (4, 4), (4, 5), (5, 4), (5, 5), (6, 6)\}$ es una relación de equivalencia en A . ¿Qué partición en A origina R ?

A.) $A_1 = \{1, 2\}$, $A_2 = \{3\}$, $A_3 = \{4, 5\}$ y $A_4 = \{6\}$

B.) $A_1 = \{1, 2\}$, $A_2 = \{3\}$ y $A_3 = \{4, 5\}$

C.) $A_1 = \{1\}$, $A_2 = \{2, 3\}$, $A_3 = \{4, 5\}$ y $A_4 = \{6\}$

D.) $A_1 = \{1\}$, $A_2 = \{2\}$, $A_3 = \{3\}$, $A_4 = \{4\}$, $A_5 = \{5\}$ y $A_6 = \{6\}$

E.) $A_1 = \{1\}$, $A_2 = \{1, 2, 3\}$ y $A_3 = \{4, 5, 6\}$

5. Para $A = \{1, 2, 3, 4, 5, 6\}$, $R = \{(1, 1), (1, 2), (2, 1), (2, 2), (3, 3), (4, 4), (4, 5), (5, 4), (5, 5), (6, 6)\}$ es una relación de equivalencia en A . ¿Cuál es la clase de equivalencia del 4 según la relación de equivalencia R ?

A.) $[4]_R = \{1, 2\}$

B.) $[4]_R = \{3\}$

C.) $[4]_R = \{3, 4\}$

D.) $[4]_R = \{4\}$

E.) $[4]_R = \{4, 5\}$

Capítulo 9

Introducción a la teoría de números

9.1. Los números enteros y la división

9.1.1. Introducción

Las matemáticas discretas involucran a los números enteros y sus propiedades dentro de un campo que se llama Teoría de Números.

9.1.2. División entre números enteros

Cuando un número entero es dividido por un segundo número entero (diferente de cero), el resultado puede ser o no ser un entero. Por ejemplo $\frac{10}{2} = 5$ es un número entero, mientras que $\frac{13}{2} = 6,5$ no lo es. Esto conduce a la siguiente definición.

Definición: Si a y b son números enteros con $a \neq 0$, se dice que a divide a b si hay un número entero c tal que $b = ac$. Cuando a divide a b se dice que a es un factor de b y que b es un múltiplo de a . La notación $a|b$ denota que a divide b . Se escribe $a \nmid b$ cuando a no divide b .

Ejemplo 1:

Determinar si $4|9$ y si $4|20$.

Se tiene que $4 \nmid 9$ porque $\frac{9}{4} = 2,25$, y $2,25$ no es un número entero.

Es cierto que $4|20$ porque existe el número $c = 5$ el cual hace que $20 = 4 \cdot 5$.

Ejemplo 2:

Sean n y d enteros positivos. ¿Cuántos enteros positivos no exceden a n y son divisibles por d ?

Los números enteros positivos divisibles por d son todos los enteros de la forma $d \cdot k$, donde k es un entero positivo. Por lo tanto, el número de enteros positivos divisibles por d que no exceden a n es igual al número de enteros k con $0 < d \cdot k \leq n$, o con $0 < k \leq \frac{n}{d}$, por lo tanto, hay un total de $\lfloor \frac{n}{d} \rfloor$ números enteros positivos que no exceden a n y que son divisibles por d .

Algunas de las propiedades básicas de divisibilidad de los números enteros son presentadas en el siguiente teorema.

Teorema:

Sean a , b y c números enteros. Entonces:

1. Si $a \mid b$ y $a \mid c$ entonces $a \mid (b + c)$
2. Si $a \mid b$ entonces $a \mid bc$ para todo entero c
3. Si $a \mid b$ y $b \mid c$ entonces $a \mid c$
4. Si $a \mid b$ y $a \mid c$ entonces $a \mid (mb + nc)$ donde m y n son enteros.

Para demostrar cada uno de los ítems del teorema se hará uso de la técnica de demostración directa y de la definición de división entre números enteros, donde se tiene:

1. Si $a \mid b$ y $a \mid c$ entonces $a \mid (b + c)$.

Si $a \mid b$ es porque existe un número entero p tal que $b = a \cdot p$, si $a \mid c$ es porque existe un número q tal que $c = a \cdot q$, se tiene que $b + c = a \cdot p + a \cdot q$, $b + c = a \cdot (p + q)$, por lo tanto a partir de la definición de división entre números enteros se tiene que $a \mid (b + c)$ porque $b + c = a \cdot (p + q)$.

2. Si $a \mid b$ entonces $a \mid bc$ para todo entero c .

Tenemos que si $a \mid b$ es porque existe un número entero p tal que $b = a \cdot p$, de este modo se tiene que $a \mid bc$ puede transformarse en $a \mid a \cdot p \cdot c$ con lo cual se obtiene que $b \cdot c$ es múltiplo de a por lo tanto $b \cdot c$ puede dividirse por a .

3. Si $a \mid b$ y $b \mid c$ entonces $a \mid c$.

Si $b \mid c$ es porque existe un número entero p tal que $c = b \cdot p$, como se tiene que $a \mid b$ entonces existe un número entero q tal que $b = a \cdot q$, reemplazando el valor de b en $c = b \cdot p$ se tiene que $c = a \cdot q \cdot p$ donde c es un múltiplo de a , por lo tanto $a \mid c$.

4. Si $a \mid b$ y $a \mid c$ entonces $a \mid (mb + nc)$ donde m y n son enteros.

Si $a \mid b$ es porque existe un número entero p tal que $b = a \cdot p$, si $a \mid c$ es porque existe un número q tal que $c = a \cdot q$, entonces $m \cdot b + n \cdot c = m \cdot a \cdot p + n \cdot a \cdot q = a \cdot (m \cdot p + n \cdot q)$ de donde se obtiene que $m \cdot b + n \cdot c$ es múltiplo de a por lo tanto $a \mid (mb + nc)$.

9.1.3. El algoritmo de la división entre números enteros

Cuando un número entero es dividido por un número entero positivo, hay un cociente y un residuo. En el Algoritmo de la División sea a un número entero y d un número entero positivo, entonces hay unos únicos enteros q y r , con $0 \leq r < d$, tal que $a = d \cdot q + r$. En la igualdad, d es llamado el divisor, a es llamado el dividendo, q es llamado el cociente y r es llamado el residuo.

La notación utilizada es:

- $q = a \text{ div } d$, el operador *div* sirve para calcular la *parte entera* de la división entre números enteros.
- $r = a \text{ mod } d$, el operador *mod* sirve para calcular el *residuo* de la división entre números enteros.

Ejemplo 3:

¿Cuál es el cociente y el residuo cuando 233 es dividido por 20?

El número 233 puede obtenerse de la siguiente forma con respecto al divisor 20,

$$233 = 20 \cdot 11 + 13$$

$233 \text{ div } 20 = 11$, la cantidad de veces que se encuentra presente el 20 en el 233 es 11, el cual es el cociente, de donde se obtiene que $20 \cdot 11 = 220$.

$233 \text{ mod } 20 = 13$, el residuo de la división entera es igual a 13, el cual también se obtiene de la siguiente manera $233 - 220 = 13$.

9.1.4. Los números primos

Cada número entero positivo mayor que 1 es divisible por al menos dos números enteros. Cuando un número entero mayor que 1 es divisible únicamente por uno (1) y por él mismo entonces este número es llamado número primo.

Definición (número primo): Un número entero positivo p más grande que uno es llamado número primo si únicamente los factores de p son 1 y p . Un número entero positivo que es más grande que 1 y no es un número primo es llamado número compuesto.

Nota: El número entero n es compuesto si y únicamente si existe un número entero a tal que $a|n$ y $1 < a < n$.

Ejemplo 4:

¿Los números 8 y 11 son números primos?

Los factores del 8 son 1, 2, 4 y 8, como se obtienen más de dos factores entonces el

número 8 no es un número primo, pero 8 sí es un número compuesto.

Los únicos factores del 11 son el 1 y el 11, por lo tanto el 11 es un número primo.

9.1.5. Teorema fundamental de la aritmética

Cada número entero positivo mayor que 1 puede ser escrito únicamente con un número primo o con el producto de dos o más números primos donde los factores primos son escritos en orden creciente con respecto a su tamaño.

Ejemplos 5:

Utilizando el Teorema Fundamental de la Aritmética la representación prima de los números 100, 845, 999 y 1024 es la siguiente:

- $100 = 2 \cdot 2 \cdot 5 \cdot 5 = 2^2 \cdot 5^2$
- $845 = 5 \cdot 13 \cdot 13 = 5^1 \cdot 13^2$
- $999 = 3 \cdot 3 \cdot 3 \cdot 37 = 3^3 \cdot 37$
- $1024 = 2 \cdot 2 \cdot 2 \cdot 2 \cdot 2 \cdot 2 \cdot 2 \cdot 2 \cdot 2 \cdot 2 = 2^{10}$

Teorema:

Si n es un número entero compuesto, entonces n tiene un divisor primo menor o igual a \sqrt{n} .

Se utilizará la técnica de demostración directa para demostrar la validez del teorema.

Si n es un número compuesto, éste tiene un factor a con $1 < a < n$, por lo tanto, $n = a \cdot b$, donde a y b son números enteros positivos más grandes que 1. Se tiene que $a \leq \sqrt{n}$ o $b \leq \sqrt{n}$, si no fuera así entonces $a \cdot b > \sqrt{n} \cdot \sqrt{n} = n$. Por lo tanto, n tiene un divisor positivo que no excede a \sqrt{n} . Este divisor o es un número primo, o, es un número compuesto, donde por el Teorema Fundamental de la Aritmética dicho divisor tiene un divisor primo. En cualquiera de los dos casos, n tiene un divisor primo que es menor o igual a \sqrt{n} .

Del teorema anterior se tiene que un número entero n es primo si este no es divisible por algún primo menor o igual a la raíz cuadrada de n .

Ejemplo 6:

¿El número entero 169 es un número primo?.

Utilizando el teorema anterior se tiene que si 169 es un número compuesto entonces tendrá un factor primo menor o igual a $\sqrt{169}$, como $\sqrt{169} = 13$ entonces dicho factor primo es el número 13, porque $13 * 13 = 13^2 = 169$. Por lo tanto el número 169 es un número compuesto lo que le impide ser un número primo.

Ejemplo 7:

¿El número entero 641 es un número primo?.

Utilizando de nuevo el teorema anterior se tiene que si 641 es un número compuesto entonces tendrá un factor primo menor o igual a $\sqrt{641} \cong 25,318$. Los únicos factores primos que son menores o iguales a 25.318 son: 2, 3, 5, 7, 11, 13, 17, 19 y 23, pero, el número 641 no es divisible por ninguno de estos factores primos, por lo tanto, el número 641 no es un número compuesto, pero, 641 si es un número primo.

El siguiente algoritmo está “inspirado ” en el teorema anterior para determinar si un número es primo.

Booleano EsNumeroPrimo(n : Entero Positivo)

1. $d = 1$
2. $p = 3$
3. *Si ($n = 2$ o $n = 3$) Entonces*
4. *Retornar Cierto*
5. *Fin Si*
6. *Si ($n > 3$) Entonces*
7. *Si ($n \bmod 2 = 0$) Entonces*
8. $d = d + 1$
9. *Fin Si*
10. *Hacer Mientras ($p \leq \lfloor \sqrt{n} \rfloor$ y $d = 1$)*
11. *Si ($n \bmod p = 0$) Entonces*
12. $d = d + 1$
13. *Fin Si*
14. $p = p + 2$
15. *Fin Hacer Mientras*
16. *Si ($d = 1$) Entonces*
17. *Retornar Cierto*
18. *Fin Si*
19. *De Otro Modo*
20. *Retornar Falso*
21. *Fin Si*

Ejemplo 8:

Utilizar el algoritmo “EsNumeroPrimo” para determinar si el número entero 169 es un número primo?.

Al realizar el paso a paso del algoritmo se inicializan los valores de las variables d , p y n en:

$$d = 1$$

$$p = 3$$

$$n = 169$$

como 169 es mayor que 3 entonces el algoritmo sigue en la línea 7, donde el residuo de dividir 169 en 2 no es cero, por lo tanto el algoritmo sigue al ciclo de repetición “hacer mientras” de la línea 10, durante el cual siempre se tiene el siguiente valor:

$$\lfloor \sqrt{n} \rfloor = \lfloor \sqrt{169} \rfloor = \lfloor 13 \rfloor = 13$$

Iteración 1:

Se cumple la condición del ciclo de repetición hacer mientras porque el $3 \leq 13$ y $d = 1$. El residuo de dividir 169 en 3 no es cero, por lo tanto no se incrementa el valor de la variable d .

$$p = p + 2 = 3 + 2 = 5$$

Iteración 2:

Se cumple la condición del ciclo de repetición hacer mientras porque el $5 \leq 13$ y $d = 1$. El residuo de dividir 169 en 5 no es cero, por lo tanto no se incrementa el valor de la variable d .

$$p = p + 2 = 5 + 2 = 7$$

Iteración 3:

Se cumple la condición del ciclo de repetición hacer mientras porque el $7 \leq 13$ y $d = 1$. El residuo de dividir 169 en 7 no es cero, por lo tanto no se incrementa el valor de la variable d .

$$p = p + 2 = 7 + 2 = 9$$

Iteración 4:

Se cumple la condición del ciclo de repetición hacer mientras porque el $9 \leq 13$ y $d = 1$. El residuo de dividir 169 en 9 no es cero, por lo tanto no se incrementa el valor de la variable d .

$$p = p + 2 = 9 + 2 = 11$$

Iteración 5:

Se cumple la condición del ciclo de repetición hacer mientras porque el $11 \leq 13$ y $d = 1$. El residuo de dividir 169 en 11 no es cero, por lo tanto no se incrementa el valor de la variable d .

$$p = p + 2 = 11 + 2 = 13$$

Iteración 6:

Se cumple la condición del ciclo de repetición hacer mientras porque el $13 \leq 13$ y $d = 1$.

El residuo de dividir 169 en 13 es cero, por lo tanto

$$d = d + 1 = 1 + 1 = 2$$

$$p = p + 2 = 13 + 2 = 15$$

Iteración 7:

No se cumple la condición del ciclo de repetición hacer mientras ya sea porque $15 \not\leq 13$ o porque $d \neq 1$. El algoritmo continua en el condicional de la línea 16.

Por último:

Como $d \neq 1$ entonces no se cumple dicho condicional y la función termina retornando el “falso” de la línea 20, lo que indica que el número 169 no es un número primo.

Ejemplo 9:

¿El número entero 61 es un número primo?.

Al realizar el paso a paso del algoritmo se inicializan los valores de las variables d , p y n en:

$$d = 1$$

$$p = 3$$

$$n = 61$$

como 61 es mayor que 3 entonces el algoritmo sigue en la línea 7, donde el residuo de dividir 61 en 2 no es cero, por lo tanto el algoritmo sigue al ciclo de repetición “hacer mientras” de la línea 10, durante el cual siempre se tiene el siguiente valor:

$$\lfloor \sqrt{n} \rfloor = \lfloor \sqrt{61} \rfloor = \lfloor 7,810249676 \rfloor = 7$$

Iteración 1:

Se cumple la condición del ciclo de repetición hacer mientras porque el $3 \leq 7$ y $d = 1$.

El residuo de dividir 61 en 3 no es cero, por lo tanto no se incrementa el valor de la variable d .

$$p = p + 2 = 3 + 2 = 5$$

Iteración 2:

Se cumple la condición del ciclo de repetición hacer mientras porque el $5 \leq 7$ y $d = 1$.

El residuo de dividir 61 en 5 no es cero, por lo tanto no se incrementa el valor de la variable d .

$$p = p + 2 = 5 + 2 = 7$$

Iteración 3:

Se cumple la condición del ciclo de repetición hacer mientras porque el $7 \leq 7$ y $d = 1$. El residuo de dividir 61 en 7 no es cero, por lo tanto no se incrementa el valor de la variable d .

$$p = p + 2 = 7 + 2 = 9$$

Iteración 4:

No se cumple la condición del ciclo de repetición hacer mientras porque $9 \not\leq 7$. El algoritmo continua en el condicional de la línea 16.

Por último:

Como $d = 1$ entonces se cumple dicho condicional y la función termina retornando “cierto”, lo que indica que el número 61 si es un número primo.

9.1.6. Procedimiento para generar la factorización prima de un número entero

Gracias al Teorema Fundamental de la Aritmética, se garantiza que todo número entero n mayor que uno tiene una única factorización prima, dicha factorización prima se puede determinar con el siguiente algoritmo:

Procedimiento FactorizacionPrima(n : Entero Positivo)

1. $i = 0$
2. $p = 2$
3. Si ($n > 1$) Entonces
4. Hacer Mientras ($p \leq \lfloor \sqrt{n} \rfloor$)
5. Si ($n \bmod p = 0$) Entonces
6. $a_i = p$
7. $n = n \text{ div } p$
8. $i = i + 1$
9. Fin Si
10. De Otro Modo
11. $p = \text{GenerarSiguientePrimo}(p)$
12. Fin Hacer Mientras
13. $a_i = n$
14. Fin Si

La factorización prima del número n es $a_0 \cdot a_1 \cdot a_2 \cdot \dots \cdot a_k$, donde se obtienen k factores primos no necesariamente diferentes, donde $a_0 \leq a_1 \leq a_2 \leq \dots \leq a_k$.

El procedimiento hace uso de la función “GenerarSiguientePrimo”, la cual se define a continuación, dicha función lo que hace es generar el siguiente número que se obtiene a partir de un número primo p .

Entero Positivo GenerarSiguientePrimo(p: Entero Positivo)

1. *Si* ($p = 2$) *Entonces*
2. *Retornar* 3
3. *Fin Si*
4. *De Otro Modo Si* ($p > 2$) *Entonces*
5. $m = p + 2$
6. *Hacer Mientras* ($EsNumeroPrimo(m) \neq Cierto$)
7. $m = m + 2$
8. *Fin Hacer Mientras*
9. *Retornar* m
10. *Fin De Otro Modo*

Ejemplo 10:

Utilizar el algoritmo para generar la factorización prima del número entero 100.

Al realizar el paso a paso del algoritmo (o prueba de escritorio) se inicializan los valores de las variables i , p y n en:

$$i = 0$$

$$p = 2$$

$$n = 100$$

como 100 es mayor que 1 entonces el algoritmo sigue en el ciclo de repetición.

Iteración 1:

$$\lfloor \sqrt{n} \rfloor = \lfloor \sqrt{100} \rfloor = \lfloor 10 \rfloor = 10$$

Se cumple la condición del ciclo de repetición hacer mientras porque el $2 \leq 10$.

El residuo de dividir 100 en 2 es cero, por lo tanto:

$$a_i = a_0 = 2$$

$$n = \frac{100}{2} = 50$$

$$i = i + 1 = 0 + 1 = 1$$

Iteración 2:

$$\lfloor \sqrt{n} \rfloor = \lfloor \sqrt{50} \rfloor = \lfloor 7,071067812 \rfloor = 7$$

Se cumple la condición del ciclo de repetición hacer mientras porque el $2 \leq 7$.

El residuo de dividir 50 en 2 es cero, por lo tanto:

$$a_i = a_1 = 2$$

$$n = \frac{50}{2} = 25$$

$$i = i + 1 = 1 + 1 = 2$$

Iteración 3:

$$\lfloor \sqrt{n} \rfloor = \lfloor \sqrt{25} \rfloor = \lfloor 5 \rfloor = 5$$

Se cumple la condición del ciclo de repetición hacer mientras porque el $2 \leq 5$.

El residuo de dividir 25 en 2 no es cero, por lo tanto:

$$p = \text{GenerarSiguientePrimo}(2) = 3$$

Iteración 4:

$$\lfloor \sqrt{n} \rfloor = \lfloor \sqrt{25} \rfloor = \lfloor 5 \rfloor = 5$$

Se cumple la condición del ciclo de repetición hacer mientras porque el $3 \leq 5$.

El residuo de dividir 25 en 3 no es cero, por lo tanto:

$$p = \text{GenerarSiguientePrimo}(3) = 5$$

Iteración 5:

$$\lfloor \sqrt{n} \rfloor = \lfloor \sqrt{25} \rfloor = \lfloor 5 \rfloor = 5$$

Se cumple la condición del ciclo de repetición hacer mientras porque el $5 \leq 5$.

El residuo de dividir 25 en 5 es cero, por lo tanto:

$$a_i = a_2 = 5$$

$$n = \frac{25}{5} = 5$$

$$i = i + 1 = 2 + 1 = 3$$

Iteración 6:

$$\lfloor \sqrt{n} \rfloor = \lfloor \sqrt{5} \rfloor = \lfloor 2,236067978 \rfloor = 2$$

No se cumple la condición del ciclo de repetición y se sale de él porque el $5 \not\leq 2$.

Por último:

$$a_i = a_3 = n = 5$$

La factorización prima del número cien, es $100 = a_0 \cdot a_1 \cdot a_2 \cdot a_3 = 2 \cdot 2 \cdot 5 \cdot 5 = 2^2 \cdot 5^2$.

Ejemplo 11:

Utilizar el algoritmo para generar la factorización prima del número entero 641.

Al realizar el paso a paso del algoritmo se inicializan los valores de las variables i , p y n en:

$$i = 0$$

$$p = 2$$

$$n = 641$$

como 641 es mayor que 1 entonces el algoritmo sigue en el ciclo de repetición.

Iteración 1:

$$\lfloor \sqrt{n} \rfloor = \lfloor \sqrt{641} \rfloor = \lfloor 25,3179778 \rfloor = 25$$

Se cumple la condición del ciclo de repetición hacer mientras porque el $2 \leq 25$.

El residuo de dividir 641 en 2 no es cero, por lo tanto:

$p = \text{GenerarSiguientePrimo}(2) = 3$

Iteración 2:

$$\lfloor \sqrt{n} \rfloor = \lfloor \sqrt{641} \rfloor = \lfloor 25,3179778 \rfloor = 25$$

Se cumple la condición del ciclo de repetición hacer mientras porque el $3 \leq 25$.

El residuo de dividir 641 en 3 no es cero, por lo tanto:

$p = \text{GenerarSiguientePrimo}(3) = 5$

Iteración 3:

$$\lfloor \sqrt{n} \rfloor = \lfloor \sqrt{641} \rfloor = \lfloor 25,3179778 \rfloor = 25$$

Se cumple la condición del ciclo de repetición hacer mientras porque el $5 \leq 25$.

El residuo de dividir 641 en 5 no es cero, por lo tanto:

$p = \text{GenerarSiguientePrimo}(5) = 7$

Iteración 4:

$$\lfloor \sqrt{n} \rfloor = \lfloor \sqrt{641} \rfloor = \lfloor 25,3179778 \rfloor = 25$$

Se cumple la condición del ciclo de repetición hacer mientras porque el $7 \leq 25$.

El residuo de dividir 641 en 7 no es cero, por lo tanto:

$p = \text{GenerarSiguientePrimo}(7) = 11$

Iteración 5:

$$\lfloor \sqrt{n} \rfloor = \lfloor \sqrt{641} \rfloor = \lfloor 25,3179778 \rfloor = 25$$

Se cumple la condición del ciclo de repetición hacer mientras porque el $11 \leq 25$.

El residuo de dividir 641 en 11 no es cero, por lo tanto:

$p = \text{GenerarSiguientePrimo}(11) = 13$

Iteración 6:

$$\lfloor \sqrt{n} \rfloor = \lfloor \sqrt{641} \rfloor = \lfloor 25,3179778 \rfloor = 25$$

Se cumple la condición del ciclo de repetición hacer mientras porque el $13 \leq 25$.

El residuo de dividir 641 en 13 no es cero, por lo tanto:

$p = \text{GenerarSiguientePrimo}(13) = 17$

Iteración 7:

$$\lfloor \sqrt{n} \rfloor = \lfloor \sqrt{641} \rfloor = \lfloor 25,3179778 \rfloor = 25$$

Se cumple la condición del ciclo de repetición hacer mientras porque el $17 \leq 25$.

El residuo de dividir 641 en 17 no es cero, por lo tanto:

$p = \text{GenerarSiguientePrimo}(17) = 19$

Iteración 8:

$$\lfloor \sqrt{n} \rfloor = \lfloor \sqrt{641} \rfloor = \lfloor 25,3179778 \rfloor = 25$$

Se cumple la condición del ciclo de repetición hacer mientras porque el $19 \leq 25$.

El residuo de dividir 641 en 19 no es cero, por lo tanto:

$$p = \text{GenerarSiguientePrimo}(19) = 23$$

Iteración 9:

$$\lfloor \sqrt{n} \rfloor = \lfloor \sqrt{641} \rfloor = \lfloor 25,3179778 \rfloor = 25$$

Se cumple la condición del ciclo de repetición hacer mientras porque el $23 \leq 25$.

El residuo de dividir 641 en 23 no es cero, por lo tanto:

$$p = \text{GenerarSiguientePrimo}(23) = 29$$

Iteración 10:

$$\lfloor \sqrt{n} \rfloor = \lfloor \sqrt{641} \rfloor = \lfloor 25,3179778 \rfloor = 25$$

No se cumple la condición del ciclo de repetición hacer mientras porque el $29 \not\leq 25$.

Por último:

$$a_i = a_0 = n = 641$$

La factorización prima es $641 = a_0 = 641 = 641^1$.

Teorema:

Los números primos son infinitos.

Se probará este teorema utilizando la técnica de demostración por contradicción. Se asume que existe una cantidad finita de números primos, los cuales son: $p_1, p_2, p_3, \dots, p_n$. Sea $Q = p_1 \cdot p_2 \cdot p_3 \cdot \dots \cdot p_n + 1$. Por el Teorema Fundamental de la Aritmética, Q es un número primo o de otro modo puede ser escrito como el producto de dos o más números primos. Sin embargo, ninguno de los primos p_j divide a Q , por lo tanto Q es un número primo y se llega a una contradicción con respecto a que el conjunto de los números primos sea un conjunto finito. De esta forma queda demostrado que el conjunto de los números primos es infinito.

9.1.7. El máximo común divisor (MCD)

Sean a y b números enteros diferentes de cero. El entero más grande d tal que $d|a$ y $d|b$ es llamado el Máximo Común Divisor (MCD) de a y b . El máximo común divisor de a y b se denota por $MCD(a, b)$.

Ejemplo 12:

¿Cuál es el $MCD(24, 36)$?

Los divisores comunes de ambos números son: 1, 2, 3, 4, 6 y 12, donde

$$MCD(24, 36) = MAX(\{1, 2, 3, 4, 6, 12\}) = 12.$$

Ejemplo 13:

¿Cuál es el $MCD(17, 22)$?

$$MCD(17, 22) = 1.$$

Definición (primos relativos): Los números enteros a y b son primos relativos si el MCD es igual a 1.

Ejemplo 14:

17 y 22 son primos relativos porque $MCD(17, 22) = 1$.

Otra forma de encontrar el MCD de dos números enteros es usando la factorización prima de los dos números a y b , donde:

$$a = p_1^{a_1} p_2^{a_2} \dots p_n^{a_n}, \quad b = p_1^{b_1} p_2^{b_2} \dots p_n^{b_n}$$

se debe tener en cuenta que los exponentes son números naturales.

El Máximo Común Divisor de los números enteros a y b se obtiene de la siguiente forma:

$$MCD(a, b) = p_1^{\min(a_1, b_1)} p_2^{\min(a_2, b_2)} \dots p_n^{\min(a_n, b_n)}.$$

Ejemplo 15:

La factorización prima de 120 y 500 es: $120 = 2^3 \cdot 3^1 \cdot 5^1$, $500 = 2^2 \cdot 3^0 \cdot 5^3$

$$MCD(120, 500) = 2^{\min(3, 2)} \cdot 3^{\min(1, 0)} \cdot 5^{\min(1, 3)} = 2^2 \cdot 3^0 \cdot 5^1 = 20$$

9.1.8. El mínimo común múltiplo(MCM)

El Mínimo Común Múltiplo de los números enteros positivos a y b es el número entero positivo más pequeño que es divisible por ambos números enteros a y b . El mínimo común múltiplo entre a y b es denotado por $MCM(a, b)$.

La forma más fácil de calcular el MCM entre dos números enteros es utilizando la factorización prima de dichos números, donde:

$$a = p_1^{a_1} p_2^{a_2} \dots p_n^{a_n}, \quad b = p_1^{b_1} p_2^{b_2} \dots p_n^{b_n}$$

se debe tener en cuenta que los exponentes son números naturales.

El Mínimo Común Múltiplo de los números enteros a y b se obtiene de la siguiente forma:

$$MCM(a, b) = p_1^{\max(a_1, b_1)} p_2^{\max(a_2, b_2)} \dots p_n^{\max(a_n, b_n)}.$$

Ejemplo 16:

¿Cuál es el $MCM(95256, 432)$?

La factorización prima de 95256 y 432 es: $95256 = 2^3 \cdot 3^5 \cdot 7^2$, $432 = 2^4 \cdot 3^3$

$$MCM(95256, 432) = 2^{\max(3, 4)} \cdot 3^{\max(5, 3)} \cdot 7^{\max(2, 0)} = 2^4 \cdot 3^5 \cdot 7^2 = 16 \cdot 243 \cdot 49 = 190512$$

Ejemplo 17:

Sean los números enteros 50 y 25, $MCD(50, 25) = 25$ y $MCM(50, 25) = 50$, $50 \cdot 25 = MCD(50, 25) \cdot MCM(50, 25) = 25 \cdot 50$

9.1.9. El algoritmo de Euclides

La siguiente es una versión iterativa del algoritmo de Euclides para calcular el máximo común divisor de dos números enteros positivos:

Procedimiento $MCD(a, b : \text{Enteros Positivos})$

1. $x = a$
2. $y = b$
3. *Hacer Mientras* ($y \neq 0$)
4. $r = x \bmod y$
5. $x = y$
6. $y = r$
7. *Fin Hacer Mientras*

en el algoritmo $x \bmod y$ es el residuo de la división entera de x por y .

Al finalizar el algoritmo el máximo común divisor de los números enteros x y y se encuentra almacenado en la variable x .

La siguiente es una versión recursiva del algoritmo de Euclides para calcular el máximo común divisor de dos números enteros positivos:

$$Euclides(x, y) = \begin{cases} y & \text{si } y \leq x \text{ y } (x \bmod y) = 0 \\ Euclides(y, x) & \text{si } x < y \\ Euclides(y, (x \bmod y)) & \text{de otro modo} \end{cases}$$

Lema (en el que se apoya el algoritmo de Euclides): Sea $a = bq + r$, donde a , b , q y r son enteros. Entonces $MCD(a, b) = MCD(b, r)$.

Ejemplo 18:

Encontrar el $MCD(662, 414)$ usando el algoritmo de Euclides.

$$\begin{aligned} Euclides(662, 414) &= Euclides(414, (662 \bmod 414)) = Euclides(414, 248) = \\ &Euclides(248, (414 \bmod 248)) = Euclides(248, 166) = \\ &Euclides(166, (248 \bmod 166)) = Euclides(166, 82) = \\ &Euclides(82, (166 \bmod 82)) = Euclides(82, 2) = 2 \end{aligned}$$

9.2. Aritmética modular

En ocasiones lo único que interesa es el residuo de un número al ser dividido por otro. Por ejemplo, ¿qué hora será dentro de 100 horas a partir de este momento?, sabemos que dentro de 24 horas será exactamente la misma hora actual, lo mismo sucede dentro de 48, 72 y 96 horas, donde tendremos exactamente la misma hora actual, como simplemente faltan 4 horas para completar las 100 horas, entonces dentro de 100 horas tendremos exactamente la misma hora actual más cuatro horas.

Definición (congruencia): Si a y b son números enteros y m es un número entero positivo, entonces a es congruente a b modulo m si m divide a $(a - b)$. Se usa la notación $a \equiv b(\bmod m)$ para indicar que a es congruente a b modulo m . Si a y b no son congruentes modulo m , se escribe $a \not\equiv b(\bmod m)$.

Teorema (congruencia):

Sea a y b números enteros y m un entero positivo. Entonces $a \equiv b(\bmod m)$ si y únicamente si $a \bmod m = b \bmod m$.

Ejemplo 19:

Determinar si 17 es congruente a 5 modulo 6, es decir, ¿ $17 \equiv 5(\bmod 6)$?

Utilizando la definición de congruencia se tiene que el 6 divide a $17 - 5 = 12$, es decir $6|(17 - 5) = 6|(12) = 6|(6 \cdot 2) = 2$, de donde se concluye que 17 es congruente a 5 modulo 6.

Utilizando el teorema de congruencia se tiene $17 \bmod 6 = 5 \bmod 6$, $5 = 5$, como se presenta la igualdad entonces es cierto que 17 es congruente a 5 modulo 6.

Ejemplo 20:

Determinar si 24 es congruente a 14 modulo 6, es decir, ¿ $24 \equiv 14(\bmod 6)$?

Utilizando la definición de congruencia se tiene que el 6 no divide a $24 - 14 = 10$, es decir $6 \nmid (24 - 14) = 6 \nmid (10)$, de donde se concluye que 24 no es congruente a 14 modulo 6, $24 \not\equiv 14 \pmod{6}$.

Utilizando el teorema de congruencia se tiene $24 \pmod{6} \neq 14 \pmod{6}$, $0 \neq 2$, como no se presenta la igualdad entonces 24 no es congruente a 14 modulo 6.

Teorema:

Sea m un número entero positivo. Los números enteros a y b son congruentes modulo m si y únicamente si hay un entero k tal que $a = b + k \cdot m$.

Teorema:

Sea m un número entero positivo. Si $a \equiv b \pmod{m}$ y $c \equiv d \pmod{m}$ entonces $a + c \equiv b + d \pmod{m}$ y $a \cdot c \equiv b \cdot d \pmod{m}$

Ejemplo 21:

Se tiene que $7 \equiv 2 \pmod{5}$ y que $11 \equiv 1 \pmod{5}$ entonces con respecto al teorema anterior se tiene que $7 + 11 \equiv 2 + 1 \pmod{5}$, $18 \equiv 3 \pmod{5}$ que efectivamente es cierto y que $7 \cdot 11 \equiv 2 \cdot 1 \pmod{5}$, $77 \equiv 2 \pmod{5}$ lo cual también es cierto.

9.2.1. Aplicaciones de la aritmética modular

La Aritmética Modular tiene muchas aplicaciones en matemáticas discretas y ciencias de la computación. Algunas de la aplicaciones más importantes son las siguientes:

- Asignación de localizaciones de memoria en el computador.
- Generación de números pseudoaleatorios.
- Criptosistemas basados en aritmética modular.

9.2.2. Asignación de localizaciones de memoria en el computador

Para la asignación de localizaciones de memoria en el computador comúnmente son utilizadas las Funciones Hash (que al buscar en el español una traducción adecuada serían las “funciones resumen”). Las funciones hash son utilizadas cuando el dominio de los elementos que se van a almacenar en el computador es muy grande y la cantidad de elementos a guardar es muy poca o es muy pequeña con respecto al dominio de los elementos, tal es el caso de las nuevas cédulas de ciudadanía en Colombia que utilizan números mayores a mil millones y donde los colombianos no somos más que cincuenta millones de habitantes.

Sea $h(k)$ la función hash definida como $h(k) = k \bmod m$, donde k es la llave que representa al registro (en nuestro ejemplo k es el número de cédula de una persona y donde la cédula se considera como llave principal porque permite diferenciar de forma única a cualquier colombiano con respecto a todos los colombianos) y m es la cantidad de posiciones en memoria que se tienen para almacenar los registros, perfectamente en el contexto universitario un valor adecuado de m podría ser 100000, para considerar que se puede guardar como máximo la información de 100000 estudiantes.

Ejemplo 22:

¿En que posición de memoria debería de quedar almacenada la información del estudiante de cédula 1456452525 si se considera un valor de $m = 100000$?

La información deberá quedar almacenada en el registro ubicado en la posición 52525, el cual es el resultado que se obtiene con la función hash

$$h(1456452525) = 1456452525 \bmod 100000 = 52525$$

La función h no es inyectiva porque dos llaves (cédulas) pueden hacer referencia al mismo registro para almacenar su información allí, tal es el caso de las cédulas 1456452525 y 4987152525 que hacen referencia al registro ubicado en la posición 52525, éste tipo de problemas es solucionado utilizando manejo de colisiones, las cuales están fuera del alcance de este libro.

9.2.3. Generación de números pseudoaleatorios

El procedimiento más comúnmente utilizado en los computadores para generar números pseudoaleatorios es el de Congruencia Lineal.

El procedimiento de Congruencia Lineal utiliza cuatro números enteros: el modulo m , el multiplicador a , el incremento c y la semilla x_0 , con $2 \leq a < m$, $0 \leq c < m$ y $0 \leq x_0 < m$. Se genera una secuencia de números pseudoaleatorios $\{x_n\}$, con $0 \leq x_n < m$ para toda n , al usar sucesivamente la congruencia:

$$x_{n+1} = (a \cdot x_n + c) \bmod m$$

Ejemplo 23:

¿Cuál es la secuencia de números que se generan al elegir el modulo $m = 9$, el multiplicador $a = 7$, el incremento $c = 4$ y la semilla $x_0 = 3$?

$$\begin{aligned} x_0 &= 3 \\ x_1 &= (7 \cdot x_0 + 4) \bmod 9 = (7 \cdot 3 + 4) \bmod 9 = 25 \bmod 9 = 7 \\ x_2 &= (7 \cdot x_1 + 4) \bmod 9 = (7 \cdot 7 + 4) \bmod 9 = 53 \bmod 9 = 8 \\ x_3 &= (7 \cdot x_2 + 4) \bmod 9 = (7 \cdot 8 + 4) \bmod 9 = 60 \bmod 9 = 6 \end{aligned}$$

$$\begin{aligned}
x_4 &= (7 \cdot x_3 + 4) \bmod 9 = (7 \cdot 6 + 4) \bmod 9 = 46 \bmod 9 = 1 \\
x_5 &= (7 \cdot x_4 + 4) \bmod 9 = (7 \cdot 1 + 4) \bmod 9 = 11 \bmod 9 = 2 \\
x_6 &= (7 \cdot x_5 + 4) \bmod 9 = (7 \cdot 2 + 4) \bmod 9 = 18 \bmod 9 = 0 \\
x_7 &= (7 \cdot x_6 + 4) \bmod 9 = (7 \cdot 0 + 4) \bmod 9 = 4 \bmod 9 = 4 \\
x_8 &= (7 \cdot x_7 + 4) \bmod 9 = (7 \cdot 4 + 4) \bmod 9 = 32 \bmod 9 = 5 \\
x_9 &= (7 \cdot x_8 + 4) \bmod 9 = (7 \cdot 5 + 4) \bmod 9 = 39 \bmod 9 = 3
\end{aligned}$$

Donde $x_9 = x_0 = 3$ y cada término en la secuencia depende únicamente del término previo, de esta forma la secuencia generada es: 3, 7, 8, 6, 1, 2, 0, 4, 5, 3, 7, 8, 6, 1, 2, 0, 4, 5, 3, ...

9.2.4. Criptosistemas basados en aritmética modular

Una de las aplicaciones más importantes de la aritmética modular es la criptografía. Para ilustrar este tema se presentará el método como se encriptaban los mensajes en la época del Emperador Julio Cesar.

El Método de Encriptación del Emperador Julio Cesar consiste en seleccionar un alfabeto sobre el cual se va a escribir un mensaje, cada letra que conforma el mensaje original es reemplazada por la letra que se encuentra m posiciones a la derecha en el alfabeto. Se considera que el alfabeto es cíclico, es decir que después de la última letra del alfabeto sigue la primer letra del alfabeto, de esta forma siempre se puede desplazar m posiciones a la derecha del alfabeto sin importar la posición que ocupa ésta sobre dicho alfabeto.

Ejemplo 24:

Considerar que se tiene el siguiente alfabeto: {a, b, c, d, e, f, g, h, i, j, k, l, m}, y que en el encriptamiento una letra se reemplaza por la que esté tres posiciones a la derecha de ésta, en los casos en que se necesita seguir contando letras y éstas se acaben entonces se sigue contando desde la primer letra del alfabeto, es decir se debe considerar que el alfabeto es cíclico. ¿Cómo se debe escribir la palabra *magia* encriptada de esta forma?

Según el alfabeto las letras de la palabra “magia” son reemplazadas por: $m \rightarrow c$, $a \rightarrow d$, $g \rightarrow j$, $i \rightarrow l$ y $a \rightarrow d$. *Encriptar*(‘magia’) = ‘cdjld’.

Formalmente el proceso consiste en:

- Si se están utilizando todas las letras del alfabeto español entonces reemplazar cada letra por un entero entre 0 y 26, donde $a \rightarrow 0$, $b \rightarrow 1$, ..., $z \rightarrow 26$.
- Para cada una de las posiciones de las letras calcular la nueva posición apoyado en la siguiente formula $f(p) = (p + \text{desplazamiento}) \bmod 27$
- Para cada una de las nuevas posiciones, reemplazar dichas posiciones por la letra que corresponde en el alfabeto para dicha ubicación.

Ejemplo 25:

¿Cuál es el mensaje secreto producido por la palabra “universidad” teniendo un desplazamiento de 8 en el método de Julio Cesar y considerando todo el alfabeto español?

Equivalencia de las letras a sus posiciones comenzando desde la posición cero:

$a \rightarrow 0,$ $b \rightarrow 1,$ $c \rightarrow 2,$ $d \rightarrow 3,$ $e \rightarrow 4,$ $f \rightarrow 5,$ $g \rightarrow 6,$ $h \rightarrow 7,$
 $i \rightarrow 8,$ $j \rightarrow 9,$ $k \rightarrow 10,$ $l \rightarrow 11,$ $m \rightarrow 12,$ $n \rightarrow 13,$ $\tilde{n} \rightarrow 14,$ $o \rightarrow 15,$
 $p \rightarrow 16,$ $q \rightarrow 17,$ $r \rightarrow 18,$ $s \rightarrow 19,$ $t \rightarrow 20,$ $u \rightarrow 21,$ $v \rightarrow 22,$ $w \rightarrow 23,$
 $x \rightarrow 24,$ $y \rightarrow 25,$ $z \rightarrow 26.$

$ReemplazarLetrasPosiciones('universidad') = '21 \ 13 \ 8 \ 22 \ 4 \ 18 \ 19 \ 8 \ 3 \ 0 \ 3'$

Los desplazamientos para cada una de las posiciones de las letras son:

$f(21) = (21 + 8) \bmod 27 = 29 \bmod 27 = 2$
 $f(13) = (13 + 8) \bmod 27 = 21 \bmod 27 = 21$
 $f(8) = (8 + 8) \bmod 27 = 16 \bmod 27 = 16$
 $f(22) = (22 + 8) \bmod 27 = 30 \bmod 27 = 3$
 $f(4) = (4 + 8) \bmod 27 = 12 \bmod 27 = 12$
 $f(18) = (18 + 8) \bmod 27 = 26 \bmod 27 = 26$
 $f(19) = (19 + 8) \bmod 27 = 27 \bmod 27 = 0$
 $f(8) = (8 + 8) \bmod 27 = 16 \bmod 27 = 16$
 $f(3) = (3 + 8) \bmod 27 = 11 \bmod 27 = 11$
 $f(0) = (0 + 8) \bmod 27 = 8 \bmod 27 = 8$
 $f(3) = (3 + 8) \bmod 27 = 11 \bmod 27 = 11$

$PosicionesDesplazadas('21 \ 13 \ 8 \ 22 \ 4 \ 18 \ 19 \ 8 \ 3 \ 0 \ 3') =$
 $'2 \ 21 \ 16 \ 3 \ 12 \ 26 \ 0 \ 16 \ 11 \ 8 \ 11'$

$PosicionesPorLetras('2 \ 21 \ 16 \ 3 \ 12 \ 26 \ 0 \ 16 \ 11 \ 8 \ 11') = 'cupdmzaplil'$

De esta forma $Encryptar('universidad') = 'cupdmzaplil'$.

9.3. Representación de los enteros en el computador

La notación que típicamente utilizamos día a día es la de base 10, donde 913 es realmente $9 \times 10^2 + 1 \times 10^1 + 3 \times 10^0$. La notación que utilizan los computadores es base 2.

Teorema:

Sea b un entero positivo más grande que 1. Entonces si n es un número entero positivo

este puede ser representado de forma única como:

$$n = a_k b^k + a_{k-1} b^{k-1} + \cdots + a_1 b + a_0$$

donde k es un número entero no negativo, a_0, a_1, \dots, a_k son números enteros positivos menores que b y $a_k \neq 0$. La representación del número entero n en base b es $(a_k a_{k-1} a_{k-2} \cdots a_1 a_0)_b$.

Ejemplo 26:

¿Cuál es la representación decimal del número entero que tiene como representación binaria a $(10111100)_2$?

$$\begin{aligned} (10111100)_2 &= 1 \cdot 2^7 + 0 \cdot 2^6 + 1 \cdot 2^5 + 1 \cdot 2^4 + 1 \cdot 2^3 + 1 \cdot 2^2 + 0 \cdot 2^1 + 0 \cdot 2^0 \\ &= 2^7 + 2^5 + 2^4 + 2^3 + 2^2 \\ &= 128 + 32 + 16 + 8 + 4 \\ &= 188 \end{aligned}$$

Ejemplo 27:

¿Cuál es la representación decimal del número entero que tiene como representación ternaria a $(210112)_3$?

$$\begin{aligned} (210112)_3 &= 2 \cdot 3^5 + 1 \cdot 3^4 + 0 \cdot 3^3 + 1 \cdot 3^2 + 1 \cdot 3^1 + 2 \cdot 3^0 \\ &= 2 \cdot 243 + 1 \cdot 81 + 1 \cdot 9 + 1 \cdot 3 + 2 \cdot 1 \\ &= 486 + 81 + 9 + 3 + 2 \\ &= 581 \end{aligned}$$

Ejemplo 28:

¿Cuál es la representación decimal del número entero que tiene como representación en base 6 a $(310541)_6$?

$$\begin{aligned} (310541)_6 &= 3 \cdot 6^5 + 1 \cdot 6^4 + 0 \cdot 6^3 + 5 \cdot 6^2 + 4 \cdot 6^1 + 1 \cdot 6^0 \\ &= 3 \cdot 7776 + 1 \cdot 1296 + 5 \cdot 36 + 4 \cdot 6 + 1 \cdot 1 \\ &= 23328 + 1296 + 180 + 24 + 1 \\ &= 24829 \end{aligned}$$

9.3.1. Representación de números enteros en base hexadecimal

La representación de números enteros en base hexadecimal o base 16 es utilizada comúnmente en ciencias de la computación. En esta base 16 símbolos son requeridos, éstos son: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F, donde las letras de la A a la F representan respectivamente los números decimales del 10 al 15.

Ejemplo 29:

¿Cuál es la representación decimal del número entero que tiene como representación en base hexadecimal a $(BC024)_{16}$?

$$\begin{aligned}(BC024)_{16} &= B \cdot 16^4 + C \cdot 16^3 + 0 \cdot 16^2 + 2 \cdot 16^1 + 4 \cdot 16^0 \\(BC024)_{16} &= 11 \cdot 16^4 + 12 \cdot 16^3 + 0 \cdot 16^2 + 2 \cdot 16^1 + 4 \cdot 16^0 \\(BC024)_{16} &= 11 \cdot 65536 + 12 \cdot 4096 + 2 \cdot 16 + 4 \cdot 1 \\(BC024)_{16} &= 720896 + 49152 + 32 + 4 = (770084)_{10}\end{aligned}$$

9.3.2. Cambio de base de un número entero escrito en base 10

Ahora se describe un algoritmo para obtener la representación en base b de un número entero n escrito en base 10.

Utilizando el algoritmo de la división se tiene que: $n = b \cdot q_0 + a_0$, donde $0 \leq a_0 < b$, a_0 es el residuo de dividir n por b , q_0 es el cociente de dividir n por b , a_0 es el dígito situado más a la derecha en la representación del número entero n . Luego se repite de nuevo el proceso para q_0 , de esta forma se tiene: $q_0 = b \cdot q_1 + a_1$, donde $0 \leq a_1 < b$, a_1 es el segundo dígito por la derecha de la representación del número entero n en base b . El proceso continúa dividiendo el cociente sucesivamente por b , obteniendo como residuos los dígitos de la representación en base b . El proceso termina cuando se obtiene un cociente igual a cero. La representación del número entero n en base b es $(a_k a_{k-1} a_{k-2} \cdots a_1 a_0)_b$.

Ejemplo 30:

¿Cuál es la representación binaria (o representación en base 2) del número entero 188 (el cual se sobre entiende que está en base 10 ó base decimal)?

$188 = 2 \cdot 94 + 0,$	$q_0 = 94,$	$a_0 = 0$
$94 = 2 \cdot 47 + 0,$	$q_1 = 47,$	$a_1 = 0$
$47 = 2 \cdot 23 + 1,$	$q_2 = 23,$	$a_2 = 1$
$23 = 2 \cdot 11 + 1,$	$q_3 = 11,$	$a_3 = 1$
$11 = 2 \cdot 5 + 1,$	$q_4 = 5,$	$a_4 = 1$
$5 = 2 \cdot 2 + 1,$	$q_5 = 2,$	$a_5 = 1$
$2 = 2 \cdot 1 + 0,$	$q_6 = 1,$	$a_6 = 0$
$1 = 2 \cdot 0 + 1,$	$q_7 = 0,$	$a_7 = 1$

como la presentación en la base b es $(a_7a_6 \dots a_1a_0)_b$ entonces el número entero 188 tiene la representación binaria $(10111100)_2$.

Ejemplo 31:

¿Cuál es la representación en base 3 (o base ternaria) del número entero 581?

$$\begin{array}{lll} 581 = 3 \cdot 193 + 2, & q_0 = 193, & a_0 = 2 \\ 193 = 3 \cdot 64 + 1, & q_1 = 64, & a_1 = 1 \\ 64 = 3 \cdot 21 + 1, & q_2 = 21, & a_2 = 1 \\ 21 = 3 \cdot 7 + 0, & q_3 = 7, & a_3 = 0 \\ 7 = 3 \cdot 2 + 1, & q_4 = 2, & a_4 = 1 \\ 2 = 3 \cdot 0 + 2, & q_5 = 0, & a_5 = 2 \end{array}$$

como la presentación en la base b es $(a_5a_4a_3a_2a_1a_0)_b$ entonces el número entero 581 tiene la siguiente representación en base 3: $(210112)_3$.

Ejemplo 32:

¿Cuál es la representación en base 6 del número entero 24829?

$$\begin{array}{lll} 24829 = 6 \cdot 4138 + 1, & q_0 = 4138, & a_0 = 1 \\ 4138 = 6 \cdot 689 + 4, & q_1 = 689, & a_1 = 4 \\ 689 = 6 \cdot 114 + 5, & q_2 = 114, & a_2 = 5 \\ 114 = 6 \cdot 19 + 0, & q_3 = 19, & a_3 = 0 \\ 19 = 6 \cdot 3 + 1, & q_4 = 3, & a_4 = 1 \\ 3 = 6 \cdot 0 + 3, & q_5 = 0, & a_5 = 3 \end{array}$$

como la presentación en la base b es $(a_5a_4a_3a_2a_1a_0)_b$ entonces el número entero 24829 tiene la siguiente representación en base 6: $(310541)_6$.

9.3.3. Algoritmo para construir la expansión de n en base b

Procedimiento RepresentacionEnBaseB(n, b : Enteros Positivos)

1. Si $(b \geq 2)$ Entonces
2. $q = n$
3. $k = 0$
4. Hacer Mientras $(q \neq 0)$
5. $a_k = q \bmod b$
6. $q = q \div b$
7. $k = k + 1$
8. Fin Hacer Mientras
9. Fin Si

La representación del número entero n en base b es $(a_{k-1}a_{k-2}a_{k-3} \dots a_1a_0)_b$.

Ejemplo 33:

Utilizar el algoritmo “RepresentacionEnBaseB” para generar la representación en base 4 del número 531.

Al realizar el paso a paso del algoritmo se inicializan los valores de las variables n y b en:

$$n = 531$$

$$b = 4$$

como b es mayor o igual a 2 entonces el algoritmo sigue en la línea 2, donde se le asignan los siguientes valores a las variables q y k :

$$q = n = 531$$

$$k = 0$$

como la variable q tiene un valor diferente de cero, entonces el algoritmo entra al ciclo de repetición “hacer mientras” de la línea 4, las siguientes son las iteraciones que realiza el ciclo de repetición:

Iteración 1:

Se cumple la condición del ciclo de repetición hacer mientras porque $q = 531 \neq 0$, el trabajo que se realiza dentro del ciclo es el siguiente:

$$\begin{array}{lll} a_k = q \bmod b, & a_0 = 531 \bmod 4, & a_0 = 3 \\ q = q \operatorname{div} b, & q = 531 \operatorname{div} 4, & q = 132 \\ k = k + 1, & k = 0 + 1, & k = 1 \end{array}$$

Iteración 2:

Se cumple la condición del ciclo de repetición hacer mientras porque $q = 132 \neq 0$, el trabajo que se realiza dentro del ciclo es el siguiente:

$$\begin{array}{lll} a_k = q \bmod b, & a_1 = 132 \bmod 4, & a_1 = 0 \\ q = q \operatorname{div} b, & q = 132 \operatorname{div} 4, & q = 33 \\ k = k + 1, & k = 1 + 1, & k = 2 \end{array}$$

Iteración 3:

Se cumple la condición del ciclo de repetición hacer mientras porque $q = 33 \neq 0$, el trabajo que se realiza dentro del ciclo es el siguiente:

$$\begin{array}{lll} a_k = q \bmod b, & a_2 = 33 \bmod 4, & a_2 = 1 \\ q = q \operatorname{div} b, & q = 33 \operatorname{div} 4, & q = 8 \\ k = k + 1, & k = 2 + 1, & k = 3 \end{array}$$

Iteración 4:

Se cumple la condición del ciclo de repetición hacer mientras porque $q = 8 \neq 0$, el trabajo que se realiza dentro del ciclo es el siguiente:

$$\begin{array}{lll}
a_k = q \bmod b, & a_3 = 8 \bmod 4, & a_3 = 0 \\
q = q \operatorname{div} b, & q = 8 \operatorname{div} 4, & q = 2 \\
k = k + 1, & k = 3 + 1, & k = 4
\end{array}$$

Iteración 5:

Se cumple la condición del ciclo de repetición hacer mientras porque $q = 2 \neq 0$, el trabajo que se realiza dentro del ciclo es el siguiente:

$$\begin{array}{lll}
a_k = q \bmod b, & a_4 = 2 \bmod 4, & a_4 = 2 \\
q = q \operatorname{div} b, & q = 2 \operatorname{div} 4, & q = 0 \\
k = k + 1, & k = 4 + 1, & k = 5
\end{array}$$

Iteración 6:

Ya no se cumple la condición del ciclo de repetición porque la variable q es igual a cero, por lo tanto el algoritmo termina.

Por último:

Como la representación del número entero 531 en base 4 es $(a_4 a_3 a_2 a_1 a_0)_4$ entonces dicha representación es $(20103)_4$.

9.3.4. Algoritmos para operaciones de números enteros en base 2

Algoritmo para la suma de números enteros en base 2

Para el siguiente algoritmo las representaciones binarias de a y b son $(a_{n-1} a_{n-2} \dots a_1 a_0)_2$ y $(b_{n-1} b_{n-2} \dots b_1 b_0)_2$, respectivamente.

Procedimiento SumaDeNumerosEnterosEnBaseBinaria(a, b : Enteros Positivos)

1. $c = 0$
2. $n = \text{MAX}(\text{CantidadBits}(a), \text{CantidadBits}(b))$
3. Para $j = 0$ Hasta $n - 1$
4. $d = \lfloor (a_j + b_j + c)/2 \rfloor$
5. $S_j = a_j + b_j + c - 2d$
6. $c = d$
7. Fin Para
8. $S_n = c$

La representación binaria de la suma es $(S_n S_{n-1} \dots S_1 S_0)_2$

Ejemplo 34:

Utilizar el algoritmo “SumaDeNumerosEnterosEnBaseBinaria” para sumar los números enteros positivos a y b que tienen respectivamente las siguientes representaciones binarias $(1101010)_2$ y $(111100)_2$.

Al realizar el paso a paso del algoritmo se inicializan los valores de las variables c , j y n en:

$$c = 0$$

$n = 7$, donde n es la cantidad de bits del número más largo de los dos que se van a sumar, en el caso del número que tenga menos bits entonces éste se lleva a la misma longitud del otro número rellenando de tantos ceros como sean necesarios en la parte izquierda del número hasta alcanzar la longitud del número más largo, por este motivo el número b queda representado así: $(0111100)_2$.

El algoritmo entra al ciclo de repetición “Para” de la línea 3, las siguientes son las iteraciones que realiza el ciclo de repetición:

Iteración 1:

El ciclo de repetición para comienza incializando su contador de ciclo j en cero, el trabajo que se realiza dentro del ciclo es el siguiente:

$$\begin{aligned} d &= \left\lfloor \frac{a_0 + b_0 + c}{2} \right\rfloor, & d &= \left\lfloor \frac{0 + 0 + 0}{2} \right\rfloor, & d &= 0 \\ s_0 &= a_0 + b_0 + c - 2d, & s_0 &= 0 + 0 + 0 - 2 \cdot 0, & s_0 &= 0 \\ c &= d, & c &= 0 \\ j &= 1 \end{aligned}$$

Iteración 2:

Se cumple la condición del ciclo de repetición para porque $j = 1 \leq 6$, el trabajo que se realiza dentro del ciclo es el siguiente:

$$\begin{aligned} d &= \left\lfloor \frac{a_1 + b_1 + c}{2} \right\rfloor, & d &= \left\lfloor \frac{1 + 0 + 0}{2} \right\rfloor, & d &= 0 \\ s_1 &= a_1 + b_1 + c - 2d, & s_1 &= 1 + 0 + 0 - 2 \cdot 0, & s_1 &= 1 \\ c &= d, & c &= 0 \\ j &= 2 \end{aligned}$$

Iteración 3:

Se cumple la condición del ciclo de repetición para porque $j = 2 \leq 6$, el trabajo que se realiza dentro del ciclo es el siguiente:

$$\begin{aligned} d &= \left\lfloor \frac{a_2 + b_2 + c}{2} \right\rfloor, & d &= \left\lfloor \frac{0 + 1 + 0}{2} \right\rfloor, & d &= 0 \\ s_2 &= a_2 + b_2 + c - 2d, & s_2 &= 0 + 1 + 0 - 2 \cdot 0, & s_2 &= 1 \\ c &= d, & c &= 0 \\ j &= 3 \end{aligned}$$

Iteración 4:

Se cumple la condición del ciclo de repetición para porque $j = 3 \leq 6$, el trabajo que se realiza dentro del ciclo es el siguiente:

$$\begin{aligned} d &= \left\lfloor \frac{a_3 + b_3 + c}{2} \right\rfloor, & d &= \left\lfloor \frac{1 + 1 + 0}{2} \right\rfloor, & d &= 1 \\ s_3 &= a_3 + b_3 + c - 2d, & s_3 &= 1 + 1 + 0 - 2 \cdot 1, & s_3 &= 0 \\ c &= d, & c &= 1 \\ j &= 4 \end{aligned}$$

Iteración 5:

Se cumple la condición del ciclo de repetición para porque $j = 4 \leq 6$, el trabajo que se realiza dentro del ciclo es el siguiente:

$$\begin{aligned} d &= \left\lfloor \frac{a_4 + b_4 + c}{2} \right\rfloor, & d &= \left\lfloor \frac{0 + 1 + 1}{2} \right\rfloor, & d &= 1 \\ s_4 &= a_4 + b_4 + c - 2d, & s_4 &= 0 + 1 + 1 - 2 \cdot 1, & s_4 &= 0 \\ c &= d, & c &= 1 \\ j &= 5 \end{aligned}$$

Iteración 6:

Se cumple la condición del ciclo de repetición para porque $j = 5 \leq 6$, el trabajo que se realiza dentro del ciclo es el siguiente:

$$\begin{aligned} d &= \left\lfloor \frac{a_5 + b_5 + c}{2} \right\rfloor, & d &= \left\lfloor \frac{1 + 1 + 1}{2} \right\rfloor, & d &= 1 \\ s_5 &= a_5 + b_5 + c - 2d, & s_5 &= 1 + 1 + 1 - 2 \cdot 1, & s_5 &= 1 \\ c &= d, & c &= 1 \\ j &= 6 \end{aligned}$$

Iteración 7:

Se cumple la condición del ciclo de repetición para porque $j = 6 \leq 6$, el trabajo que se realiza dentro del ciclo es el siguiente:

$$\begin{aligned} d &= \left\lfloor \frac{a_6 + b_6 + c}{2} \right\rfloor, & d &= \left\lfloor \frac{1 + 0 + 1}{2} \right\rfloor, & d &= 1 \\ s_6 &= a_6 + b_6 + c - 2d, & s_6 &= 1 + 0 + 1 - 2 \cdot 1, & s_6 &= 0 \\ c &= d, & c &= 1 \\ j &= 7 \end{aligned}$$

Iteración 8:

Ya no se cumple la condición del ciclo de repetición para porque $j = 7 \not\leq 6$, por lo tanto el ciclo de repetición termina y el algoritmo continua en la línea 8.

Por último:

$s_7 = 1$. El resultado de sumar los números a y b en representación binaria es $(s_7 s_6 s_5 s_4 s_3 s_2 s_1 s_0)_2$, con respecto a los valores obtenidos en el paso a paso del algoritmo el resultado es: $(10100110)_2$.

Las representaciones en base diez de los números a , b y el resultado s es:

$$\begin{aligned} a &= (1101010)_2 = 1 \cdot 2^6 + 1 \cdot 2^5 + 0 \cdot 2^4 + 1 \cdot 2^3 + 0 \cdot 2^2 + 1 \cdot 2^1 + 0 \cdot 2^0 \\ &= 1 \cdot 64 + 1 \cdot 32 + 1 \cdot 8 + 1 \cdot 2 \\ &= 64 + 32 + 8 + 2 \\ &= 106 \end{aligned}$$

$$\begin{aligned} b &= (0111100)_2 = 0 \cdot 2^6 + 1 \cdot 2^5 + 1 \cdot 2^4 + 1 \cdot 2^3 + 1 \cdot 2^2 + 0 \cdot 2^1 + 0 \cdot 2^0 \\ &= 1 \cdot 32 + 1 \cdot 16 + 1 \cdot 8 + 1 \cdot 4 \end{aligned}$$

$$\begin{aligned}
 &= 32 + 16 + 8 + 4 \\
 &= 60
 \end{aligned}$$

$$\begin{aligned}
 s &= (10100110)_2 = 1 \cdot 2^7 + 0 \cdot 2^6 + 1 \cdot 2^5 + 0 \cdot 2^4 + 0 \cdot 2^3 + 1 \cdot 2^2 + 1 \cdot 2^1 + 0 \cdot 2^0 \\
 &= 1 \cdot 2^7 + 1 \cdot 2^5 + 1 \cdot 2^2 + 1 \cdot 2^1 \\
 &= 1 \cdot 128 + 1 \cdot 32 + 1 \cdot 4 + 1 \cdot 2 \\
 &= 128 + 32 + 4 + 2 \\
 &= 166
 \end{aligned}$$

y efectivamente $106 + 60 = 166$.

Algoritmo para la multiplicación de números enteros en base 2

Para el siguiente algoritmo las representaciones binarias de a y b son $(a_{n-1}a_{n-2} \dots a_1a_0)_2$ y $(b_{n-1}b_{n-2} \dots b_1b_0)_2$, respectivamente. Adicionalmente, c_0, c_1, \dots, c_{n-1} son los productos parciales.

Procedimiento MultiplicacionDeNumerosEnterosEnBaseBinaria(a, b : Enteros Positivos)

1. $n = CantidadBits(b)$
2. Para $j = 0$ Hasta $n - 1$
3. Si $(b_j = 1)$ Entonces
4. $c_j =$ (desplazar el número a j lugares a la izquierda)
5. De Otro Modo
6. $c_j = 0$
7. Fin Para
8. $p = 0$
9. Para $j = 0$ to $n - 1$
10. $p = p + c_j$
11. Fin Para

Al terminar el algoritmo en la variable p queda almacenado el resultado de multiplicar los números enteros a y b .

Ejemplo 35:

Utilizar el algoritmo “MultiplicacionDeNumerosEnterosEnBaseBinaria” para multiplicar los números enteros positivos a y b que tienen respectivamente las siguientes representaciones binarias $(1001)_2$ y $(1101)_2$.

Al realizar el paso a paso del algoritmo se inicia el valor de la variable n con el número entero positivo 4, porque el número b en su representación binaria tiene cuatro bits.

El algoritmo continúa en la línea 2 donde ingresa al ciclo de repetición “Para” donde se realizan las siguientes iteraciones:

Iteración 1, ciclo “para” que comienza en la línea 2:

El ciclo de repetición “para” comienza inicializando su contador de ciclo j en cero, el trabajo que se realiza dentro del ciclo es el siguiente:

como $b_0 = 1$ entonces a c_0 se le asigna el número a desplazado $j = 0$ posiciones a la izquierda, de esta forma se tiene que:

$$c_0 = (1001)_2 = 1 \cdot 2^3 + 0 \cdot 2^2 + 0 \cdot 2^1 + 1 \cdot 2^0 = 8 + 1 = 9$$

$j = j + 1 = 0 + 1 = 1$, se incrementa el contador del ciclo.

Iteración 2, ciclo “para” que comienza en la línea 2:

Se cumple la condición del ciclo de repetición “para” porque $j = 1 \leq 3$, el trabajo que se realiza dentro del ciclo es el siguiente:

como $b_1 = 0$ entonces a c_1 se le asigna el número 0 (línea 6 del algoritmo), de esta forma se tiene que $c_1 = 0$ y se incrementa el contador del ciclo, donde $j = j + 1 = 1 + 1 = 2$.

Iteración 3, ciclo “para” que comienza en la línea 2:

Se cumple la condición del ciclo de repetición “para” porque $j = 2 \leq 3$, el trabajo que se realiza dentro del ciclo es el siguiente:

como $b_2 = 1$ entonces a c_2 se le asigna el número a desplazado $j = 2$ posiciones a la izquierda, de esta forma se tiene que:

$$c_2 = (100100)_2 = 1 \cdot 2^5 + 0 \cdot 2^4 + 0 \cdot 2^3 + 1 \cdot 2^2 + 0 \cdot 2^1 + 0 \cdot 2^0 = 32 + 4 = 36$$

$j = j + 1 = 2 + 1 = 3$, se incrementa el contador del ciclo.

Iteración 4, ciclo “para” que comienza en la línea 2:

Se cumple la condición del ciclo de repetición “para” porque $j = 3 \leq 3$, el trabajo que se realiza dentro del ciclo es el siguiente:

como $b_3 = 1$ entonces a c_3 se le asigna el número a desplazado $j = 3$ posiciones a la izquierda, de esta forma se tiene que:

$$c_3 = (1001000)_2 = 1 \cdot 2^6 + 0 \cdot 2^5 + 0 \cdot 2^4 + 1 \cdot 2^3 + 0 \cdot 2^2 + 0 \cdot 2^1 + 0 \cdot 2^0 = 64 + 8 = 72$$

$j = j + 1 = 3 + 1 = 4$, se incrementa el contador del ciclo.

Iteración 5, ciclo “para” que comienza en la línea 2:

Ya no se cumple la condición del ciclo de repetición “para” porque $j = 4 \not\leq 3$, por lo tanto el ciclo de repetición termina y el algoritmo continua en la línea 8 donde se inicializa la variable p con el valor cero.

El algoritmo continúa en la línea 9 donde ingresa al ciclo de repetición “Para” donde se realizan las siguientes iteraciones:

Iteración 1, ciclo “para” que comienza en la línea 9:

El ciclo de repetición “para” comienza inicializando su contador de ciclo j en cero, el trabajo que se realiza dentro del ciclo es el siguiente:

$$\begin{array}{lll} p = p + c_0, & p = 0 + 9, & p = 9 \\ j = j + 1, & j = 0 + 1 = 1, & j = 1 \end{array}$$

Iteración 2, ciclo “para” que comienza en la línea 9:

Se cumple la condición del ciclo de repetición “para” porque $j = 1 \leq 3$, el trabajo que se realiza dentro del ciclo es el siguiente:

$$\begin{array}{lll} p = p + c_1, & p = 9 + 0, & p = 9 \\ j = j + 1, & j = 1 + 1 = 1, & j = 2 \end{array}$$

Iteración 3, ciclo “para” que comienza en la línea 9:

Se cumple la condición del ciclo de repetición “para” porque $j = 2 \leq 3$, el trabajo que se realiza dentro del ciclo es el siguiente:

$$\begin{array}{lll} p = p + c_2, & p = 9 + 36, & p = 45 \\ j = j + 1, & j = 2 + 1 = 1, & j = 3 \end{array}$$

Iteración 4, ciclo “para” que comienza en la línea 9:

Se cumple la condición del ciclo de repetición “para” porque $j = 3 \leq 3$, el trabajo que se realiza dentro del ciclo es el siguiente:

$$\begin{array}{lll} p = p + c_3, & p = 45 + 72, & p = 117 \\ j = j + 1, & j = 3 + 1 = 1, & j = 4 \end{array}$$

Iteración 5, ciclo “para” que comienza en la línea 9:

Ya no se cumple la condición del ciclo de repetición “para” porque $j = 4 \not\leq 3$, por lo tanto el ciclo de repetición termina y el algoritmo también termina obteniéndose como resultado de la multiplicación el número almacenado en la variable p , cuyo valor es 117.

Las representaciones en base diez de los números a y b es:

$$\begin{aligned} a &= (1001)_2 = 1 \cdot 2^3 + 0 \cdot 2^2 + 0 \cdot 2^1 + 1 \cdot 2^0 \\ &= 1 \cdot 8 + 1 \cdot 1 \\ &= 8 + 1 \\ &= 9 \end{aligned}$$

$$\begin{aligned} b &= (1101)_2 = 1 \cdot 2^3 + 1 \cdot 2^2 + 0 \cdot 2^1 + 1 \cdot 2^0 \\ &= 1 \cdot 8 + 1 \cdot 4 + 1 \cdot 1 \\ &= 8 + 4 + 1 \\ &= 13 \end{aligned}$$

y efectivamente $9 + 13 = 117$.

9.4. Ejercicios

1. ¿El número entero positivo 1327 es un número primo?, demostrar o refutar utilizando las propiedades de los números primos.

2. ¿El número entero positivo 2371 es un número primo?, demostrar o refutar utilizando las propiedades de los números primos.
3. ¿El número entero positivo 3721 es un número primo?, demostrar o refutar utilizando las propiedades de los números primos.
4. ¿El número entero positivo 7231 es un número primo?, demostrar o refutar utilizando las propiedades de los números primos.
5. Escribir los siguientes números como un producto de factores primos, teniendo en cuenta que

$$p_1^{n_1} \cdot p_2^{n_2} \cdot \dots \cdot p_k^{n_k}, \quad 0 < n_i, \quad 1 \leq i \leq k, \quad p_1 < p_2 < \dots < p_k$$

- a.) 407125
 - b.) 184600
 - c.) 842570
 - d.) 945677
6. ¿Cuántos divisores tiene el número 407125?
 7. ¿Cuántos divisores tiene el número 184600?
 8. ¿Cuántos divisores tiene el número 842570?
 9. ¿Cuántos divisores tiene el número 945677?
 10. Apoyados en el Teorema Fundamental de la Aritmética y utilizando la factorización prima de los números enteros, determinar el Máximo Común Divisor (MCD) y el Mínimo Común Múltiplo (MCM) de las siguientes parejas de números enteros:
 - a.) 225 y 350
 - b.) 254 y 896
 - c.) 425 y 789
 - d.) 486 y 964
 - e.) 487 y 765
 - f.) 504 y 540
 - g.) 1576, 8748 y 99500
 - h.) 1976, 4258 y 80275
 - i.) 6175, 8632 y 73853
 - j.) 9846, 5700 y 94567
 11. Para cada $n \in \mathbb{Z}^+$, ¿cuál es el $MCD(n, n+1)$ y $MCM(n, n+1)$?

12. Para $a, b, d \in \mathbb{Z}^+$ y $d = MCD(a, b)$, demostrar que $MCD(\frac{a}{d}, \frac{b}{d}) = 1$
13. Para $a, b, n \in \mathbb{Z}^+$, demostrar que $MCD(n \cdot a, n \cdot b) = n \cdot MCD(a, b)$
14. Sea la relación R en \mathbb{Z}^+ , donde $(a, b) \in R$ si $MCD(a, b) = 1$. ¿Es la relación R reflexiva?, ¿es la relación R simétrica?, ¿es la relación R antisimétrica?, y ¿es la relación R transitiva?
15. Apoyados en el algoritmo de Euclides determinar el Máximo Común Divisor (MCD) de las siguientes parejas de números enteros:
 - a.) 225 y 350
 - b.) 254 y 896
 - c.) 425 y 789
 - d.) 486 y 964
 - e.) 487 y 765
 - f.) 504 y 540
 - g.) 576 y 748
 - h.) 1976 y 80275
 - i.) 6175 y 73853
16. Sea $m \in \{2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12\}$. Para los valores de m presentados anteriormente, ¿cuáles serían todos los valores que servirían para que 30 y 35 sean congruentes?
17. Sea $m \in \{2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12\}$. Para los valores de m presentados anteriormente, ¿cuáles serían todos los valores que servirían para que 40 y 52 sean congruentes?
18. Sea $m \in \{2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12\}$. Para los valores de m presentados anteriormente, ¿cuáles serían todos los valores que servirían para que 47 y 63 sean congruentes?
19. Sea $m \in \{2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12\}$. Para los valores de m presentados anteriormente, ¿cuáles serían todos los valores que servirían para que 58 y 70 sean congruentes?
- 20.Cuál es el mensaje original después de desencriptar “htwwjhyt” utilizando en el encriptamiento de Julio Cesar¹ un desplazamiento de cinco?
21. ¿Cómo se representa el número $(3213310)_4$ en base 6?

¹En este caso trabajar con el alfabeto inglés, el cual es: $\{A, B, C, D, E, F, G, H, I, J, K, L, M, N, O, P, Q, R, S, T, U, V, W, X, Y, Z\}$

22. ¿Cómo se representa el número $(3403414)_5$ en base 7?
23. ¿Cómo se representa el número $(14315234)_6$ en base 9?
24. ¿Cuál es el resultado de multiplicar los números en base dos 10101111 y 10011011 ?, justificar la respuesta al desarrollar la prueba de escritorio del algoritmo.
25. ¿Cuál es el resultado de multiplicar los números en base dos 11101011 y 10101010 ?, justificar la respuesta al desarrollar la prueba de escritorio del algoritmo.
26. ¿Cuál es el resultado de multiplicar los números en base dos 10011010 y 11101011 ?, justificar la respuesta al desarrollar la prueba de escritorio del algoritmo.
27. ¿Cuál es el resultado de multiplicar los números en base dos 10110010 y 11001101 ?, justificar la respuesta al desarrollar la prueba de escritorio del algoritmo.

9.5. Preguntas tipo ECAES

1. Se tiene el número 2593 (escrito en base diez), ¿cuál es la representación de éste en base ocho?²
 - A.) 1405
 - B.) 5041
 - C.) 14050
 - D.) 2468
 - E.) 5014
2. Se tiene el número $(2467)_8$ (escrito en base ocho), ¿cuál es la representación de éste en base tres?
 - A.) $(2011121)_3$
 - B.) $(12111020)_3$
 - C.) $(4002)_3$
 - D.) $(111121)_3$
 - E.) $(1211110)_3$
3. Un número entero positivo es primo si y únicamente si éste es divisible por la unidad y por el mismo, los número enteros que no son primos son llamados números compuestos, tener en cuenta que por el Teorema Fundamental de la Aritmética se tiene que cada entero positivo más grande que 1 puede ser escrito únicamente con un primo o con el producto de dos o más primos, de ésta forma un número compuesto es el resultado de la multiplicación de dos o más primos, además, se tiene que un número entero positivo n es un número primo si éste no es divisible de forma exacta por algún número primo menor o igual a la raíz cuadrada de n .

Con respecto al enunciado anterior para probar que un número entero positivo de doce cifras es un número primo, entonces se deben buscar factores primos como máximo hasta de tamaño³ :

- A.) 100.000
- B.) 1'000.000
- C.) 10'000.000
- D.) 100'000.000

²Las preguntas 1, 2, 4 y 5 son del autor.

³Pregunta reescrita a partir de una pregunta no liberada del ECAES de Ingeniería de Sistemas del año 2006.

E.) 999.999'999.999

4. Según el Teorema Fundamental de la Aritmética, ¿cuál es la factorización prima del número 360?

A.) $2^3 \cdot 3^2 \cdot 5^1$

B.) $2^3 \cdot 3^2 \cdot 5^2$

C.) $2^2 \cdot 3^2 \cdot 5^1 \cdot 7^1$

D.) $2^1 \cdot 4^1 \cdot 3^2 \cdot 5^1$

E.) $2^2 \cdot 3^2 \cdot 7^1$

5. ¿Cuántos divisores tiene el número 360?

A.) 6

B.) 12

C.) 18

D.) 20

E.) 24

Bibliografía

- [B1988] Bustamente, Alfonso. 1988. “Elementos de Algebra en Ciencias de la Computación”. Serie de textos universitarios, Universidad ICESI, Cali.
- [C1989] Caicedo, Xavier. 1989. “Elementos de Lógica Matemática y Calculabilidad”. Universidad de los Andes, Bogotá.
- [C1970] Codd, E. F., 1970. “A Relational Model for Large Shared Data Banks”, Communications of the ACM, volumen 13, número 6, páginas 377-387.
- [CLRS2001] Cormen, T. H., Leiserson, C. E., Rivest, R. L., and Stein, C.. 2001. “Introduction to Algorithms”, second edition. The MIT Press, Cambridge, Massachusetts.
- [G1997] Grimaldi, R.. 1997. “Matemáticas Discreta y Combinatoria”, tercera edición. Addison-Wesley Iberoamericana.
- [J1997] Johnsonbaugh, R.. 1997. “Matemáticas Discretas”. Prentice Hall.
- [J2009] Jiménez Murillo, José A.. 2009. “Matemáticas para la Computación”. Alfaomega.
- [KBR1997] Kolman, B., Busby, R. C. y Ross, S.. 1997. “Estructuras de Matemáticas Discretas para la Computación”. Prentice Hall.
- [MI2010] Ministerio de Educación Nacional Republica de Colombia - Icfes. 2010. “Guía de Orientación: Examen de Estado de Calidad de la Educación Superior en Ingeniería de Sistemas (ECAES)” [online]. ICFES, Bogotá D. C..
Disponible de internet:
http://web2.icfes.gov.co/index.php?option=com_docman&task=doc_view&gid=2807
- [MIA2003] Ministerio de Educación Nacional Republica de Colombia - Icfes - Acofi. 2003. “Exámenes de Calidad de la Educación Superior, Ingeniería de Sistemas: Cuadernillos de Preguntas Primera y Segunda Sesión”. ICFES.
- [MIA2004] Ministerio de Educación Nacional Republica de Colombia - Icfes - Acofi. 2004. “Exámenes de Calidad de la Educación Superior, Ingeniería de Sistemas: Cuadernillo de Preguntas y Respuestas Segunda Sesión”

[online]. ICFES.

Disponible de internet:

http://200.14.205.63:8080/portalicfes/home_2/rec/arc_4181.pdf

http://200.14.205.63:8080/portalicfes/home_2/rec/arc_4237.xls

- [M2010] Morales Peña, Hugo Humberto. 2010. “El Algoritmo de Ordenamiento por Comparaciones Heapinsert Sort”. Quinto Congreso Colombiano de Computación (5CCC), Cartagena de Indias, Colombia.
- [R2004] Rosen, K. H.. 2004. “Matemática Discreta y sus Aplicaciones”, quinta edición. Mc Graw Hill Interamericana de España.
- [RW1990] Ross, K. A. y Wright, C. R. B.. 1990. “Matemáticas Discretas”. Prentice Hall.
- [SKS2002] Silberschatz, A., Korth, H., y Sudarshan, S.. 2002. “Fundamentos de Bases de Datos”, cuarta edición. Mc Graw Hill Interamericana de España.
- [W1964] Williams, J. W. J.. 1964. “Algorithm 232 (HEAPSORT)”. Communications of the ACM, 7:347-348.